BitStorm 2600 and GranDSLAM 4200 IP DSLAM

SNMP Reference

Document No. 2600-A2-GB22-30

March 2005

PARADYNE®

Notice

Warranty, Sales, Service, and Training Information

Contact your local sales representative, service representative, or distributor directly for any help needed. For additional information concerning warranty, sales, service, repair, installation, documentation, training, distributor locations, or Paradyne worldwide office locations, use one of the following methods:

- **Internet:** Visit the Paradyne World Wide Web site at **www.paradyne.com**. (Be sure to register your warranty at **www.paradyne.com/warranty**.)
- **Telephone:** Call our automated system to receive current information by fax or to speak with a company representative.
  - Within the U.S.A., call 1-800-870-2221
  - Outside the U.S.A., call 1-727-530-2340

Document Feedback

We welcome your comments and suggestions about this document. Please mail them to Technical Publications, Paradyne Corporation, 8545 126th Ave. N., Largo, FL 33773, or send e-mail to **userdoc@paradyne.com**. Include the number and title of this document in your correspondence. Please include your name and phone number if you are willing to provide additional clarification.

Trademarks

Table of Contents

About This Guide

Document Purpose and Intended Audience

This guide contains the information necessary to use Simple Network Management Protocol (SNMP) to configure and monitor the BitStorm 2600 and GranDSLAM 4200 IP DSLAMs.

Document Summary

Section	Description
Section 1, Introduction	Describes the format and contents of this document.
Section 2, Traps	Lists traps supported by the BitStorm 2600 and GranDSLAM 4200 IP DSLAMs.
Section 3, Ordered MIB List	Lists in order MIBs that must be loaded for full support.
Section 4, Miscellaneous Textual Conventions	Describes typographic conventions used to convey MIB constructs and object types.
Section 5, Management Information Base (MIBs)	Describes MIB support in the BitStorm 2600 and GranDSLAM 4200 IP DSLAMs.
Index	Lists MIBs and MIB objects for easy reference.

A master glossary of terms and acronyms used in Paradyne documents is available online at **www.paradyne.com**. Select *Support* → *Technical Manuals* → *Technical Glossary*.

Product-Related Documents

Complete documentation for this product is available online at **www.paradyne.com**. Select *Support* → *Technical Manuals*.

Document Number	Document Title
2600-A2-GB21	*BitStorm 2600 and GranDSLAM 4200 IP DSLAM Command Line Interface Reference*
2600-A2-GN20	*BitStorm 2600 IP DSLAM Installation Guide*
4200-A2-GN21	*GranDSLAM 4200 IP DSLAM Installation Guide*
6210-A2-GB21	*Hotwire Models 6210-A2, 6211-A2, and 6381-A2 User's Guide*
7890-A2-GB22	*GrandVIEW EMS User's Guide*

To order a paper copy of a Paradyne document, or to speak with a sales representative, please call 1-727-530-2000.

Section 1 Introduction

1. Introduction

This document identifies the MIBs and MIB objects supported in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

The document is arranged such that each MIB is a level 2 (x.y) heading under the Management Information Base (MIBs). Each MIB section has a unique header where the name of the MIB is identified on all pages applicable to that particular MIB. The MIB sections are laid out textually such that:

- The OIDs are identified for the MIB
- Any textual conventions defined in the MIB are identified
- MIB objects are listed along with their:
  - object name
  - OID
  - syntax
  - status
  - support

Something is not supported when one of the following is true:

- A whole table does not exist in the MIB tree
- A particular object does not exist in the MIB tree
- A particular enumeration for a particulare object is not available

When a particular table or object does not exist in the MIB tree, attempted access to it by a management entity will result in a failure code of **noSuchName**.

The list of MIBs that have objects supported by the BitStorm 2600 and GranDSLAM 4200 IP DSLAM can be found in the Table of Contents section of this document.

The goal of this document is to provide all the information required to develop, test, document, and support the SNMP operation of the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

MIB objects that have values that are not fully supported are identified by the left justified heading as "**Valid Object Value(s)**." Following this heading is a list of values supported. The absence of this heading implies that all possible values for a MIB object are supported.

1.1 Internet Society Copyright Statement

Parts of RFCs have been directly copied into this document. In the words of the RFC Editor (a group funded by the Internet Society), "To protect the integrity of RFC publication, all RFCs are published with an ISOC (Internet Society) copyright statement… This copyright notice was designed to ensure that Request For Comments (RFC)

documents will have the widest possible distribution." The copyright notice that follows should be taken to apply to all RFCs and excerpts thereof in this document. For actual copyright date, see the individual RFCs.

Section 2 Traps

2. Traps

This section describes the standards compliance and any special operational features/options for the SNMP Trap support. Trap Support List, Table 1 on page 3, identifies all traps supported by the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

All Traps are sent in SNMPv1 PDU formats.

Table 1. Trap Support List (1 of 5)

Trap	Description	VarBind(s)[a]	MIB	Cross Reference
authenticationFailure	A SNMP message was received that was not properly authenticated.	devAuthentication-FailureIpAddress	SNMPv2-MIB	Section 5.2.4.3, Page 28
warmStart	Power-on reset (software reset)		SNMPv2-MIB	Section 5.2.4.2, Page 28
linkDown	The link is down	ifIndex ifAdminStatus ifOperStatus	IF-MIB	Section 5.3.5.1, Page 68
linkUp	The link is up	ifIndex ifAdminStatus ifOperStatus	IF-MIB	Section 5.3.5.2, Page 68
adslAtuc-PerfLofsThreshTrap	Loss of Framing 15-minute interval threshold reached.	adslAtucPerfCurr15MinLofs adslAtucThresh15MinLofs	ADSL-LINE-MIB	Section 5.11.3.1.1, Page 209
adslAtuc-PerfLossThreshTrap	Loss of Signal 15-minute interval threshold reached.	adslAtucPerfCurr15MinLoss adslAtucThresh15MinLoss	ADSL-LINE-MIB	Section 5.11.3.1.2, Page 209
adslAtuc-PerfESsThreshTrap	Errored Second 15-minute interval threshold reached.	adslAtucPerfCurr15MinESs adslAtucThresh15MinESs	ADSL-LINE-MIB	Section 5.11.3.1.4, Page 209
adslAtuc-RateChangeTrap	The ATUCs transmit rate has changed.	adslAtucChanCurrTxRate adslAtucChanPrevTxRate	ADSL-LINE-MIB	Section 5.11.3.1.5, Page 209
adslAtucPerf-LolsThreshTrap	Loss of Link 15-minute interval threshold reached.	adslAtucPerfCurr15MinLols adslAtucThresh15MinLols	ADSL-LINE-MIB	Section 5.11.3.1.6, Page 209
adslAtucInit-FailureTrap	ATUC initialization failed.	adslAtucCurrStatus	ADSL-LINE-MIB	Section 5.11.3.1.7, Page 210
adslAturPerf-LofsThreshTrap	Loss of Framing 15-minute interval threshold reached.	adslAturPerfCurr15MinLofs adslAturThresh15MinLofs	ADSL-LINE-MIB	Section 5.11.3.2.1, Page 210
adslAtur-PerfLossThreshTrap	Loss of Signal 15-minute interval threshold reached.	adslAturPerfCurr15MinLoss adslAturThresh15MinLoss	ADSL-LINE-MIB	Section 5.11.3.2.2, Page 210

Table 1. Trap Support List (2 of 5)

Trap	Description	VarBind(s)[a]	MIB	Cross Reference
adslAturPerf-LprsThreshTrap	Loss of Power 15-minute interval threshold reached.	adslAturPerfCurr15MinLprs adslAturThresh15MinLprs	ADSL-LINE-MIB	Section 5.11.3.2.3, Page 210
adslAtur-PerfESsThreshTrap	Errored Second 15-minute interval threshold reached.	adslAturPerfCurr15MinESs adslAturThresh15MinESs	ADSL-LINE-MIB	Section 5.11.3.2.4, Page 210
adslAtur-RateChangeTrap	The ATUCs transmit rate has changed.	adslAturChanCurrTxRate adslAturChanPrevTxRate	ADSL-LINE-MIB	Section 5.11.3.2.5, Page 210
adslAtucSesL-ThreshTrap	Severely errored seconds-line 15-minute threshold reached.	adslAtucPerfCurr15MinSesL adslAtucThresh15MinSesL	ADSL-LINE-EXT-MIB	Section 5.12.5.1.2, Page 222
adslAtucUasL-ThreshTrap	Unavailable seconds-line 15-minute threshold reached.	adslAtucPerfCurr15MinUasL adslAtucThresh15MinUasL	ADSL-LINE-EXT-MIB	Section 5.12.5.1.3, Page 222
adslAturSesL-ThreshTrap	Severely errored seconds-line 15-minute threshold reached.	adslAturPerfCurr15MinSesL adslAturThresh15MinSesL	ADSL-LINE-EXT-MIB	Section 5.12.5.2.1, Page 223
adslAturUasL-ThreshTrap	Unavailable seconds-line 15-minute threshold reached.	adslAturPerfCurr15MinUasL adslAturThresh15MinUasL	ADSL-LINE-EXT-MIB	Section 5.12.5.2.2, Page 223
dsx1LineStatus-Change	The value of a dsx1LineStatus instance changed.	dsx1LineStatus dxx1LineStatusLastChange	DS1-MIB	Section 6.14.5.1, Page 371
hdsl2Shdsl-LoopAttenCrossing	Loop attenuation threshold has been reached or exceeded.	hdsl2ShdslEndpoint-CurrAtn hdsl2ShdslEndpointThresh-LoopAttenuation	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.1, Page 340
hdsl2Shdsl-SNRMarginCrossing	SNR margin threshold has been reached or exceeded.	hdsl2ShdslEndpoint-CurrSnrMgn hdsl2ShdslEndpointThresh-SNRMargin	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.2, Page 340
hdsl2ShdslPerf-ESThresh	Errored seconds threshold has been reached or exceeded.	hdsl2ShdslEndpoint-Curr15MinES hdsl2ShdslEndpointThresh-ES	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.3, Page 340
hdsl2ShdslPerf-SESThresh	Severely errored seconds threshold has been reached or exceeded.	hdsl2ShdslEndpoint-Curr15MinSES hdsl2ShdslEndpointThresh-SES	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.4, Page 340
hdsl2ShdslPerf-CRC-anomaliesThresh	CRC anomalies threshold has been reached or exceeded.	hdsl2ShdslEndpoint-Curr15MinCRCanomalies hdsl2ShdslEndpointThresh-CRCanomalies	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.5, Page 340
hdsl2ShdslPerf-LOSWSThresh	SOSW seconds threshold has been reached or exceeded.	hdsl2ShdslEndpoint-Curr15MinLOSWS hdsl2ShdslEndpointThresh-LOSWS	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.6, Page 340

Table 1. Trap Support List (3 of 5)

Trap	Description	VarBind(s)[a]	MIB	Cross Reference
hdsl2ShdslPerf-UASThresh	Unavailable seconds threshold has been reached or exceeded.	hdsl2ShdslEndpoint-Curr15MinUAS hdsl2ShdslEndpointThresh-UAS	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.7, Page 340
hdsl2ShdslSpan-InvalidNumRepeaers	A mismatch has been detected between the number of repeater/ regenerator units configured for the line and the actual number of units discovered via the EOC.	hdsl2ShdslSpanConf-NumRepeaters hdsl2ShdslStatusNumAvail-Repeaters	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.8, Page 340
hdsl2Shdsl-LoopbackFailure	An endpoint maintenance loopback command failed for the segment.	hdsl2ShdslMaint-LoopbackConfig	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.9, Page 340
hdsl2Shdsl-powerBackoff	The bit setting in the status for powerBackoff for the endpoint has been changed.	hdsl2ShdslEndpoint-CurrStatus	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.10, Page 340
hdsl2Shdsl-deviceFault	The bit setting in the status for deviceFault for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.11, Page 341
hdsl2Shdsl-dcContinuityFault	The bit setting in the status for dcContinuity-Fault for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.12, Page 341
hdsl2Shdsl-configInitFailure	The bit setting in the status for configInitFailure for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.13, Page 341
hdsl2Shdsl-protocolInitFailure	The bit setting in the status for protocolInitFailure for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.14, Page 341
hdsl2Shdsl-noNeighborPresent	The bit setting in the status for noNeighborPresent for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.15, Page 341
hdsl2Shdsl-LocalPowerLoss	Indicates an impending unit failure due to loss of local power.	hdsl2ShdslInvVendorID	HDSL2-SHDSL-LINE-MIB	Section 6.13.7.16, Page 341
pdnDev-FileXferEvent	A File Transfer Event, firmware or configuration occurred	• pdnDevFileXferStatus • pdnDevFileXferErrorStatus • pdnDevFileXferOperation • pdnDevFileXferFileType • pdnDevFileXferFileName • pdnDevFileXferErrorStatusText	PDN-CONTROL-MIB	Section 5.27.5.5, Page 390
mpeCcn	Configuration change (rate limited)	entPhysicalIndex	PDN-MPE-DSLAM-SYSTEM-MIB	Section 5.20.5.1, Page 325

Table 1. Trap Support List (4 of 5)

Trap	Description	VarBind(s)[a]	MIB	Cross Reference
mpeSelfTest-Failure	Sent if any portion of a restart/self start fails	mpeDevSelfTestResults	PDN-MPE-HEALTH-AND-STATUS-MIB	Section 5.21.5.1, Page 328
mpeAlarmRelay-InputStateChanged	The contacts on the Alarm Relay input have change from open to closed or from closed to open.	mpeAlarmRelayInputState	PDN-MPE-DSLAM-SYSTEM-MIB	Section 5.20.5.10, Page 326
pdnEntPhySensor ExtThreshold-ExceededSet	A sensor that was previously operating within a defined threshold is now operating in excess of the threshold limits.	• pdnPhySensorValue • pdnEntPhySensorExt-ThresholdState	PDN-ENTITY-SENSOR-EXT-MIB	Section 5.30.5.1, Page 412
pdnEntPhySensor-ExtThreshold-ExceededCleared	A sensor that was previously operating in excess of a defined threshold is now operating within the threshold limits.	• pdnPhySensorValue • pdnEntPhySensorExt-ThresholdState	PDN-ENTITY-SENSOR-EXT-MIB	Section 5.30.5.2, Page 412
pdnHdsl2Shdsl-LoopAttenCrossing	Loop attenuation threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-CurrAtn • hdsl2ShdslEndpointThresh-LoopAttenuation	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.1, Page 758
pdnHdsl2Shdsl-SNRMarginCrossing	SNR margin threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-CurrSnrMgn • hdsl2ShdslEndpointThresh-SNRMargin	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.2, Page 759
pdnHdsl2ShdslPerf-ESThresh	Errored seconds threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-Curr15MinES • hdsl2ShdslEndpointThresh-ES	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.3, Page 759
pdnHdsl2ShdslPerf-SESThresh	Severely errored seconds threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-Curr15MinSES • hdsl2ShdslEndpointThresh-SES	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.4, Page 759
pdnHdsl2ShdslPerf-CRC-anomaliesThresh	CRC anomalies threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-Curr15MinCRCanomalies • hdsl2ShdslEndpointThresh-CRCanomalies	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.5, Page 759
pdnHdsl2ShdslPerf-LOSWSThresh	SOSW seconds threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-Curr15MinLOSWS • hdsl2ShdslEndpointThresh-LOSWS	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.6, Page 759
pdnHdsl2ShdslPerf-UASThresh	Unavailable seconds threshold has been reached or exceeded.	• hdsl2ShdslEndpoint-Curr15MinUAS • hdsl2ShdslEndpointThresh-UAS	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.7, Page 759
pdnHdsl2Shdsl-powerBackoff	The bit setting in the status for powerBackoff for the endpoint has been changed.	hdsl2ShdslEndpoint-CurrStatus	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.8, Page 759

Table 1. Trap Support List (5 of 5)

Trap	Description	VarBind(s)[a]	MIB	Cross Reference
pdnHdsl2Shdsl-deviceFault	The bit setting in the status for deviceFault for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.9, Page 759
pdnHdsl2Shdsl-dcContinuityFault	The bit setting in the status for dcContinuity-Fault for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.10, Page 759
pdnHdsl2Shdsl-configInitFailure	The bit setting in the status for configInitFailure for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.11, Page 759
pdnHdsl2Shdsl-protocolInitFailure	The bit setting in the status for protocolInitFail-ure for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.12, Page 759
pdnHdsl2Shdsl-noNeighborPresent	The bit setting in the status for noNeighborPre-sent for the endpoint has changed.	hdsl2ShdslEndpoint-CurrStatus	PDN-HDSL2-SHDSL-LINE-MIB	Section 6.55.5.13, Page 760

a. The snmpTrapOID and snmpTrapEnterprise objects apply only to SNMPv2-Trap-PDUs and InformRequest-PDUs and will not be supported in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM as it those PDU types are not used to generate trap/notifications.

Section 3 Ordered MIB List

3. Ordered MIB List

This section contains an ordered list the MIBs and their dependencies supported in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. In other words, MIBs that are dependent on other MIBs are listed after the MIBs they are dependent on. The objective here is to list MIBs in the order in which they should be loaded in an NMS application. This list is "worst case" in that it contains all possible MIBs that might be required, although not all of them may be required. A lot of what is required is dependent on the needs of a particular NMS application. This list was constructed by examining the "IMPORTS" clause in the MIB and making sure those items MIBs identified were listed before the particular MIB that was importing them. MIBs in **bold** font are the MIBs containing objects implemented in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. Those that are not in bold font are dependencies.

Table 2. Ordered MIB List (1 of 3)

MIB	RFC/File
SNMPv2-SMI	RFC 2578
SNMPv2-CONF	RFC 2580
SNMPv2-TC	RFC 2579
SNMPv2- MIB	**RFC 1907**
RFC1213-MIB	**RFC 1213**
IANAifType-MIB	http://www.iana.org/assignments/ianaiftype.mib
PDN-HEADER-MIB	pdn_Header.mib
PDN-TC	pdn_tc.mib
SNMP-FRAMEWORK-MIB	RFC 2571
IP-MIB	**RFC 2011**
ENTITY-MIB	**RFC 2737**
ENTITY-SENSOR-MIB	**RFC 3433**
IF-MIB	**RFC 2863**
IP-FORWARD-MIB	**RFC 2096**
ATM-TC-MIB	RFC 2514
ATM-MIB	**RFC 2515**
HOST-RESOURCES-MIB	RFC 2790
ATM-FORUM-SNMP-M4-MIB	**af-nm-0095.001.mib**
RS-232-MIB	**RFC 1659**

Table 2. Ordered MIB List (2 of 3)

MIB	RFC/File
EtherLike-MIB	**RFC 2665**
MAU-MIB	**RFC 2668**
PerfHist-TC-MIB	RFC 2493
ADSL-TC-MIB, **ADSL-LINE-MIB**	**RFC 2662**
ADSL-LINE-EXT-MIB	**RFC 3440**
BRIDGE-MIB	**RFC 1493**
RFC1158-MIB	RFC 1158
RFC1271-MIB	RFC 1271
RMON-MIB	RFC 2819
TOKEN-RING-RMON-MIB[a]	RFC 1513
RMON2-MIB	RFC 2021
P-BRIDGE-MIB **Q-BRIDGE-MIB**	**RFC 2674**
INET-ADDRESS-MIB	RFC 2851
IGMP-STD-MIB	**RFC 2933**
HDSL2-SHDSL-LINE-MIB	**RFC 3276**
DS1-MIB	**RFC 2495**
PPP-LCP-MIB	**RFC 1471**
PPP-BRIDGE-NCP-MIB	**RFC 1474**
PDN-HEADER-MIB	pdn_Header.mib
PDN-TC	pdn_tc.mib
PDN-MPE-DEVICE-CONTROL-MIB	**mpe_Control.mib**
PDN-MPE-HEALTH-AND-STATUS-MIB	**mpe_HealthAndStatus.mib**
PDN-MPE-MIB2-MIB	mpe_mib2.mib
PDN-MPE-DSLAM-SYSTEM-MIB	**mpe_dslam.mib**
PDN-ARP-MIB	**pdn_Arp.mib**
PDN-ATMSTATS-MIB	**pdn_AtmStats.mib**
PDN-CONFIG-MIB	**pdn_Config.mib**
PDN-CONTROL-MIB	**pdn_Control.mib**
PDN-IFEXT-MIB	pdn_IfExt.mib
PDN-IF-EXT-CONFIG-MIB	**pdn_IfExtConfig.mib**
PDN-SECURITY-MIB	**pdn_Security.mib**
PDN-DIAGNOSTICS-MIB	**pdn_diag.mib**

Table 2. Ordered MIB List (3 of 3)

MIB	RFC/File
PDN-DSLAM-SYSTEM-MIB	**pdn_dslam.mib**
PDN-ETHER-MIB	**pdn_ether.mib**
PDN-FILTER-MIB	**pdn_filter.mib**
PDN-INET-MIB	**pdn_inet.mib**
PDN-SYSLOG-MIB	**pdn_syslog.mib**
PDN-UPLINK-TAGGING-MIB	**PDN-UPLINK-TAGGING-MIB.mib**
PDN-STACKABLE-MIB	**PDN-STACKABLE-MIB.mib**
PDN-DEVICE-TIME-MIB	**pdn_time.mib**
PDN-ATM-BRIDGE-IWF-MIB	**PDN-ATM-BRIDGE-IWF-MIB.mib**
PDN-REACHDSL-MIB	**pdn_reachDSL.mib**
PDN-SPECTRUMMGMT-MIB	**pdn_spectrummgmt.mib**
PDN-ATMEXT-MIB	**pdn_AtmExt.mib**
PDN-ATMM4EXT-MIB	**pdn_AtmM4Ext.mib**
PDN-ENTITY-SENSOR-EXT-MIB	**PDN-ENTITY-SENSOR-EXT-MIB.mib**
PDN-VLAN-MIB	**PDN-VLAN-MIB.mib**
PDN-IGMP-STD-EXT-MIB	**PDN-IGMP-STD-EXT-MIB.mib**
PDN-IP-MULTICAST-MIB	**PDN-IP-MULTICAST-MIB.mib**
PDN-HDSL2-SHDSL-LINE-MIB	**PDN-HDSL2-SHDSL-LINE-MIB.mib**
PDN-ADSL-LINE-EXT-MIB	**PDN-ADSL-LINE-EXT-MIB.mib**
PDN-DOT1QEXT-MIB	**PDN-DOT1QEXT-MIB.mib**
PDN-DS1CONFIG-MIB	**pdn_Ds1Config.mib**
PDN-DS1EXT-MIB	**PDN-DS1EXT-MIB.mib**
PDN-DHCP-MIB	**PDN-DHCP-MIB.mib**
PDN-HDLC-MIB	**PDN-HDLC-MIB.mib**
PDN-IFDEV-IWF-MIB	**PDN-IFDEV-IWF-MIB.mib**
PDN-PPP-LCP-EXT-MIB	**PDN-PPP-LCP-EXT-MIB.mib**
PDN-PPP-BRIDGE-NCP-EXT-MIB	**PDN-PPP-BRIDGE-NCP-EXT-MIB.mib**
PDN-PPP-ML-MIB	**PDN-PPP-ML-MIB.mib**
PDN-BRIDGE-EXT-MIB	**PDN-BRIDGE-EXT-MIB.mib**
PDN-HEALTHANDSTATUS-MIB	pdn_HealthAndStatus.mib[b]

a. Modified the TOKEN-RING-RMON-MIB, RFC 1513, to import the RMON-MIB instead of RFC1271-MIB as there were serious issues that would have to be worked out to get the RFC1271-MIB to compile.

b. The PDN-HEALTHANDSTATUS-MIB exists only to support the devAuthenticationFailureIpAddress variable bind for the authenticationFailure trap/notification.

Section 4 Textual Conventions

4. Miscellaneous Textual Conventions

This sections contains Textual Conventions reference by MIBs contained in this SNMP Operational Specification, but not defined in MIBs contained in this specification.

4.1 BITS

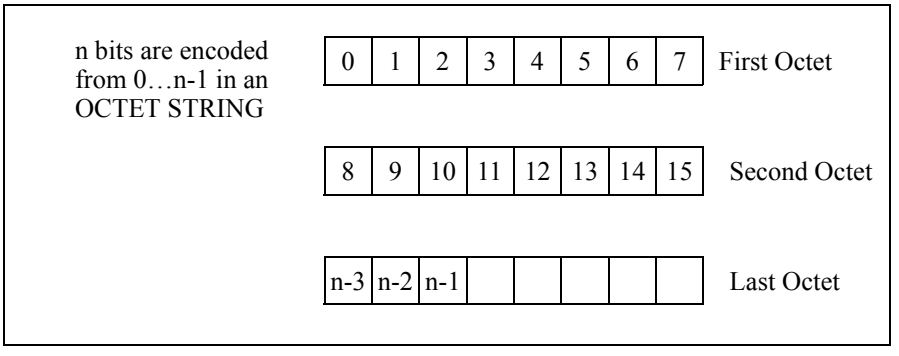
SMIv2 — RFC 2578

The BITS construct represents an enumeration of named bits. This collection is assigned non-negative, contiguous (but see below) values, starting at zero. Only those named-bits so enumerated may be present in a value.

As part of updating an information module, for an object defined using the BITS construct, new enumerations can be added or existing enumerations can have new labels assigned to them. After an enumeration is added, it might not be possible to distinguish between an implementation of the updated object for which the new enumeration is not asserted, and an implementation of the object prior to the addition. Depending on the circumstances, such an ambiguity could either be desirable or could be undesirable. The means to avoid such an ambiguity is dependent on the encoding of values on the wire; however, one possibility is to define new enumerations starting at the next multiple of eight bits. (Of course, this can also result in the enumerations no longer being contiguous.)

Although there is no SMI-specified limitation on the number of enumerations (and therefore on the length of a value), except as may be imposed by the limit on the length of an OCTET STRING, MIB designers should realize that there may be implementation and interoperability limitations for sizes in excess of 128 bits.

Finally, a label for a named-number enumeration must consist of one or more letters or digits, up to a maximum of 64 characters, and the initial character must be a lower-case letter. (However, labels longer than 32 characters are not recommended.) Note that hyphens are not allowed by this specification.



Encoding of BITS TC

4.2 ContactState

PDN-TC

This convention can be used to indicate the state of a switch, door, connector, etc. that can be by nature opened or closed.

SYNTAXINTEGER

- open(1)
- close(2)

4.3 PerfCurrentCount

PerfHist-TC-MIB — RFC 3593

A counter associated with a performance measurement in a current 15 minute measurement interval. The value of this counter starts from zero and is increased when associated events occur, until the end of the 15 minute interval. At that time the value of the counter is stored in the first 15 minute history interval, and the CurrentCount is restarted at zero. In the case where the agent has no valid data available for the current interval the corresponding object instance is not available and upon a retrieval request a corresponding error message shall be returned to indicate that this instance does not exist (for example, a noSuchName error for SNMPv1 and a noSuchInstance for SNMPv2 GET operation).

SYNTAXGauge32

4.4 PerfIntervalCount

PerfHist-TC-MIB — RFC 3593

A counter associated with a performance measurement in a previous 15 minute measurement interval. In the case where the agent has no valid data available for a particular interval the corresponding object instance is not available and upon a retrieval request a corresponding error message shall be returned to indicate that this instance does not exist (for example, a noSuchName error for SNMPv1 and a noSuchInstance for SNMPv2 GET operation). In a system supporting a history of n intervals with IntervalCount(1) and IntervalCount(n) the most and least recent intervals respectively, the following applies at the end of a 15 minute interval:

- discard the value of IntervalCount(n)
- the value of IntervalCount(i) becomes that of IntervalCount(i-1) for n >= i > 1
- the value of IntervalCount(1) becomes that of CurrentCount
- the TotalCount, if supported, is adjusted.

SYNTAXGauge32

4.5 PdnPPPState

PDN-TC — pdn_tc.mib

The state of the PPP Entity State Machine. Examples of PPP entities are Link Control Protocol (LCP), Network Control Protocol (NCP), and Multilink PPP (ML PPP).

Reference: [RFC 1661] Simpson, W., *The Point-to-Point Protocol*, Section 4.2, “State Transition Table,” July 1994.

SYNTAXINTEGER

- initial(1)
- starting(2)
- closed(3)
- stopped(4)
- closing(5)
- stopping(6)
- reqSent(7)
- ackRcvd(8)
- ackSent(9)

- opened(10)

4.6 RowStatus

SNMPv2-TC — RFC 2579

The RowStatus textual convention is used to manage the creation and deletion of conceptual rows, and is used as the value of the SYNTAX clause for the status column of a conceptual row.

SYNTAXINTEGER

- active(1) - Indicates that the conceptual row is available for use by the managed device.This is a desired state that can be written and read.
- notInService(2) - Indicates that the conceptual row exists in the agent, but is unavailable for use by the managed device. This is a desired state that can be written and read.
- notReady(3) - Indicates that the conceptual row exists in the agent, but is missing information necessary in order to be available for use by the managed device (i.e., one or more required columns in the conceptual row have not been instantiated). This is a current state and can only be read.
- createAndGo(4) - Supplied by a management station wishing to create a new instance of a conceptual row and to have its status automatically set to active, making it available for use by the managed device. This is an action state that can be written but not read.
- ~~createAndWait~~(5) - Supplied by a management station wishing to create a new instance of a conceptual row (but not make it available for use by the managed device). This is an action state that can be written but not read. This state is generally **not** supported for creating rows into tables. The createAndGo(4) state should be used to create rows in table.
- destroy(6) - Supplied by a management station wishing to delete all of the instances associated with an existing conceptual row.

4.7 SwitchState

PDN-TC — pdn_tc.mib

This object indicates whether an object state is enabled or disabled.

SYNTAXINTEGER

- enabled(1)
- disabled(2)

4.8 TimeFilter

RMON2-MIB — RFC 2021

To be used for the index to a table. Allows an application to download only those rows changed since a particular time. A row is considered changed if the value of any object in the row changes or if the row is created or deleted.

When sysUpTime is equal to zero, this table shall be empty.

One entry exists for each past value of sysUpTime, except that the whole table is purged should sysUpTime wrap.

As this basic row is updated new conceptual rows are created (which still share the now updated object values with all other instances). The number of instances which are created is determined by the value of sysUpTime at which the basic row was last updated. One instance will exist for each value of sysUpTime at the last update time for the row. A new timeMark instance is created for each new sysUpTime value. Each new conceptual row will be associated with the timeMark instance which was created at the value of sysUpTime with which the conceptual row is to be associated.

By definition all conceptual rows were updated at or after time zero and so at least one conceptual row (associated with timeMark.0) must exist for each underlying (basic) row.

Consider the following fooTable:

```
fooTable …
INDEX { fooTimeMark, fooIndex }
FooEntry {
                fooTimeMarkTimeFilter
                fooIndexINTEGER,
                fooCountsCounter
}
```

Should there be two basic rows in this table (fooIndex == 1, fooIndex == 2) and row 1 was updated most recently at time 6, while row 2 was updated most recently at time 8, and both rows had been updated on several earlier occasions such that the current values were 5 and 9 respectively then the following fooCounts instances would exist.

```
fooCounts.0.15
fooCounts.0.29
fooCounts.1.15
fooCounts.1.29
fooCounts.2.15
fooCounts.2.29
fooCounts.3.15
fooCounts.3.29
fooCounts.4.15
fooCounts.4.29
fooCounts.5.15
fooCounts.5.29
fooCounts.6.15
fooCounts.6.29
fooCounts.7.29   -- note that row 1 does not exist for
fooCounts.8.29   -- times 7 and 8
```

SYNTAXTimeTicks

4.9 TimeInterval

A period of time, measured in units of 0.01 seconds.

SYNTAX INTEGER (0…2147483647)

4.10 TimeStamp

SNMPv2-TC — RFC 2579

The value of the sysUpTime object at which a specific occurrence happened. The specific occurrence must be defined in the description of any object defined using this type.

If sysUpTime is reset to zero as a result of a reinitialization of the network management (sub)system, then the values of all TimeStamp objects are also reset. However, after approximately 497 days without a re- initialization, the sysUpTime object will reach 2^{32-1} and then increment around to zero; in this case, existing values of TimeStamp objects do not change. This can lead to ambiguities in the value of TimeStamp objects.

SYNTAXTimeTicks

4.11 TimeTicks

SMIv2 — RFC 2578

The TimeTicks type represents a non-negative integer which represents the time, modulo 2^{32} (4294967296 decimal), in hundredths of a second between two epochs.

When objects are defined which use this ASN.1 type, the description of the object identifies both of the reference epochs.

SYNTAX IMPLICIT INTEGER (0…4294967295)

4.12 TruthValue

SNMPv1-TC — RFC 2579

Represents a boolean value.

SYNTAXINTEGER

- true(1)
- false(2)

Section 5 MIBs

5. Management Information Base (MIBs)

This section describes the SNMP MIB support implemented in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM project. MIBs supported consist of both “standard” MIBs and Paradyne Enterprise MIBs.

Unlike the “standard” MIBs, the Paradyne Enterprise MIBs change frequently. As a result, the objects listed for these MIBs in this document may be a subset of the latest version of the particular enterprise MIB. Objects that exist in the later versions but not in this document are not supported.

5.1 RFC1213-MIB — RFC 1213

5.1.1 Introduction

This MIB defines objects for managing devices in TCP/IP based internets. The RFC used as a reference was dated March 1991.

5.1.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- system - { mib-2 1 } 1.3.6.1.2.1.1
- interfaces - { mib-2 2 } 1.3.6.1.2.1.2
- at - { mib-2 3 } 1.3.6.1.2.1.3
- ip - { mib-2 4 } 1.3.6.1.2.1.4
- icmp - { mib-2 5 } 1.3.6.1.2.1.5
- tcp - { mib-2 6 } 1.3.6.1.2.1.6
- udp - { mib-2 7 } 1.3.6.1.2.1.7
- egp - { mib-2 8 } 1.3.6.1.2.1.8
- cmot - { mib-2 9 } 1.3.6.1.2.1.9
- transmission - { mib-2 10 } 1.3.6.1.2.1.10
- snmp - { mib-2 11 } 1.3.6.1.2.1.11

5.1.3 MIB Objects

The following groups supported in this MIB are as follow:

Table 3. RFC1213-MIB Objects

Group	Supported
system[a]	Y
interfaces[b]	Y
at	N
ip[c]	Y
icmp[c]	N
tcp[d]	N
udp[e]	N
egp	N
cmot	N
transmission	Y
snmp[f]	Y

a. Per RFC 1907
b. Per RFC 2863
c. Per RFC 2011
d. Per RFC 2012
e. Per RFC 2013
f. Per RFC 1907

5.1.3.1 System Group

This group has been updated in SNMPv2-MIB — RFC 1907, Section 5.2, Page 21.

5.1.3.2 Interfaces Group

This group has been updated in IF-MIB — RFC 2863, Section 5.3, Page 29.

5.1.3.3 IP Group

This group has been updated in IP-MIB — RFC 2011IP-MIB, Section 5.4, Page 69.

5.1.3.4 TCP Group

Not supported.

5.1.3.5 UDP Group

Not supported.

5.1.3.6 EGP Group

Not supported.

5.1.3.6.1 Transmission Group

The Transmission Group is based on the transmission media underlying each interface on a system. The objects in the Transmission Group are not defined within MIB-II, but rather through other internet-standard MIB definitions. Objects that are part of this group can be derived from the OIDs documented in the relevant RFCs. The following describes the structure to parse for in the relevant RFCs:

type OBJECT IDENTIFIER := { transmission *number* }

Where:

— *type* - Is the symbolic value used for the media in the ifType column of the ifTable object
See Interfaces Group for ifType values and the ifTable object

— *number* - The actual integral value corresponding to the *type*

5.1.3.7 SNMP Group

This group had been updated in RFC 1907. Please refer to that section of the document for support.

5.2 SNMPv2-MIB — RFC 1907

5.2.1 Introduction

This MIB defines managed object which describe the behavior of a SNMPv2 entity. The RFC used as a reference was dated January 1996. The LAST-UPDATED for the snmp-MIB was “9511090000Z.”

5.2.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- system - { mib-2 1 } 1.3.6.1.2.1.1
- snmp - { mib-2 11 } 1.3.6.1.2.1.1.11

- snmpV2 - { internet 6 } 1.3.6.1.6
- snmpModules - { snmpV2 3 } 1.3.6.1.6.3
- snmpMIB - { snmpModules 1 } 1.3.6.1.6.3.1
- snmpMIBObjects - { snmpMIB 1 } 1.3.6.1.6.3.1.1
- snmpTrap - { snmpMIBObjects 4 } 1.3.6.1.6.3.1.1.4
- snmpTrapOID - { snmpTrap 1 } 1.3.6.1.6.3.1.1.4.1
- snmpTrapEnterprise - { snmpTrap 2 } 1.3.6.1.6.3.1.1.4.2
- snmpTraps - { snmpMIBObjects 5 } 1.3.6.1.6.3.1.1.5
- snmpSet - { snmpMIBObjects 6 } 1.3.6.1.6.3.1.1.6

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2
- pdn-devStatus - { pdn-common 4 } 1.3.6.1.4.1.1795.2.24.2.4
- devStatus - { pdn-devStatus 1 } 1.3.6.1.4.1.1795.2.24.2.4.1
- devAuthenticationFailureIpAddress - { devStatus 5 } 1.3.6.1.4.1.1795.2.24.2.4.1.5

5.2.3 MIB Objects

The table below identifies the objects called out in this MIB. Any table object that has elements that are supported are indicated by a “Y” in the Supported column.

Table 4. SNMPv2-MIB Objects

Object	OID	Supported
system	{ mib-2 1 }	Y
snmp	{ mib-2 11 }	Y

5.2.3.1 System Group

A collection of objects common to all managed systems.

OIDs:

- system - { mib-2 1 } 1.3.6.1.2.1.1

Table 5. System Group

Object	OID	Syntax	Access	Status	Supported
sysDescr	{ system 1 }	DisplayString	read-only	current	Y
sysObjectID	{ system 2 }	OBJECT IDENTIFIER	read-only	current	Y
sysUpTime	{ system 3 }	TimeTicks	read-only	current	Y
sysContact	{ system 4 }	DisplayString(SIZE(0….32))[a]	read-write	current	Y
sysName	{ system 5 }	DisplayString(SIZE(0….32))[a]	read-write	current	Y
sysLocation	{ system 6 }	DisplayString(SIZE(0….32))[a]	read-write	current	Y
sysServices	{ system 7 }	INTEGER(0…127)	read-only	current	Y
sysORLastChange	{ system 8 }	TimeStamp	read-only	current	N
sysORTable	{ system 9 }	Sequence of sysOREntry	not-accessible	current	N

a. Product imposed limit.

5.2.3.1.1 sysDescr

A textual description of the entity. This value should include the full name and version identification of the system's hardware type, software operating-system, and networking software.

Valid Object Value(s)

This object provides the full name and version identification for the systems hardware and software. This object is set to display a string with the following format:

"*Company Name*[1] *Family*[2] *Model Number*[3]; S/W Release: yy.yy.yy;"

Where:

— "yy.yy.yy" represents the software revision number

Note: Information that is unique on a "per slice" basis is reported in the Entity MIB.

5.2.3.1.2 sysObjectID

The vendor's authoritative identification of the network management subsystem contained in the entity. This value is allocated within the SMI enterprises subtree (1.3.6.1.4.1) and provides an easy and unambiguous means for determining "what kind of box" is being managed. For example, if vendor "Flintstones, Inc." was assigned the subtree 1.3.6.1.4.1.4242, it could assign the identifier 1.3.6.1.4.1.4242.1.1 to its "Fred Router."

Valid Object Value(s)

1. *Company Name* is set to "PARADYNE."
2. The *Family* is "BitStorm" for the 26xx products and "GranDSLAM" for the 42xx products.
3. The *Model Number* is "2600" for the BitStorm family products and "4200" for the GranDSLAM family products.

Below is an OID tree for sysObjectIDs for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

- *Company Enterprise*[1] .1.14.17 - ip-stackable
- *Company Enterprise*[1] .1.14.17.1 - ips-stack
- *Company Enterprise*[1] .1.14.17.1.2 - ips-2600
- *Company Enterprise*[1] .1.14.17.1.3 - ips-4200

Given that the product is part of the ip-stackable family of products, but is managed independently (that is, there is no single management entity), this object will return:

***Company Enterprise OID*[1].1.14.17.1.2**

5.2.3.1.3 sysUpTime

The time (in hundredths of a second) since the network management portion of the system was last re-initialized.

5.2.3.1.4 sysContact

The textual identification of the contact person for this managed node, together with information on how to contact this person. If no contact information is known, the value is the zero-length string.

5.2.3.1.5 sysName

An administratively-assigned name for this managed node. By convention, this is the node's fully-qualified domain name. If the name is unknown, the value is the zero-length string.

Valid Object Value(s)

Spaces are not allowed in the string.

5.2.3.1.6 sysLocation

The physical location of this node (for example, “telephone closet, 3rd floor”). If the location is unknown, the value is the zero-length string.

5.2.3.1.7 sysServices

A value which indicates the set of services that this entity may potentially offers. The value is a sum. This sum initially takes the value zero, Then, for each layer, L, in the range 1 through 7, that this node performs transactions for, 2 raised to (L - 1) is added to the sum. For example, a node which performs only routing functions would have a value of 4 (2^(3-1)). In contrast, a node which is a host offering application services would have a value of 72 (2^(4-1) + 2^(7-1)). Note that in the context of the Internet suite of protocols, values should be calculated accordingly:

For systems including OSI protocols, layers 5 and 6 may also be counted.

Valid Object Value(s)

The table below contains the set of services and their associated values based on the standard $2^{(L-1)}$ formula. The final result for this object is 1 + 2 + 4 + 8 + 64 = 79 = 0x4F.

Table 6. sysServices

Layer (L)	Functionality	Value ($2^{(L-1)}$)	Sum
1	Physical	$2^{(1-1)} = 2^0 = 1$	1
2	Data Link / Subnetwork	$2^{(2-1)} = 2^1 = 2$	3
3	Internet	$2^{(3-1)} = 2^2 = 4$	7

1. *Company Enterprise OID* is set to “1.3.6.1.4.1.1795.”

Table 6. sysServices

Layer (L)	Functionality	Value ($2^{(L-1)}$)	Sum
4	End-to-End	$2^{(4-1)} = 2^3 = 8$	15
7	Application	$2^{(7-1)} = 2^6 = 64$	**79**

5.2.3.1.8 Object Resource Information

A collection of objects which describe the SNMPv2 entity's (statically and dynamically configurable) support of various MIB modules.

5.2.3.1.8.1 sysORLastChange

The value of sysUpTime at the time of the most recent change in state or value of any instance of sysORID.

5.2.3.1.8.2 sysORTable

The (conceptual) table listing the capabilities of the local SNMPv2 entity acting in an agent role with respect to various MIB modules. SNMPv2 entities having dynamically-configurable support of MIB modules will have a dynamically-varying number of conceptual rows.

Index:

- sysORIndex

OIDs:

- sysORTable - { system 9 } 1.3.6.1.2.1.1.9
- sysOREntry - { sysORTable 1 } 1.3.6.1.2.1.1.9.1

Table 7. sysORTable

Object	OID	Syntax	Access	Status	Supported
sysORIndex	{ sysOREntry 1 }	INTEGER (1…2147483647)	not-accessible	current	N
sysORID	{ sysOREntry 2 }	OBJECT IDENTIFIER	read-only	current	N
sysORDescr	{ sysOREntry 3 }	DisplayString	read-only	current	N
sysORUpTime	{ system 4 }	TimeStamp	read-only	current	N

5.2.3.1.8.2.1 sysORIndex

The auxiliary variable used for identifying instances of the columnar objects in the sysORTable.

5.2.3.1.8.2.2 sysORID

An authoritative identification of a capabilities statement with respect to various MIB modules supported by the local SNMPv2 entity acting in an agent role.

5.2.3.1.8.2.3 sysORDescr

A textual description of the capabilities identified by the corresponding instance of sysORID.

5.2.3.1.8.2.4 sysORUpTime

The value of sysUpTime at the time this conceptual row was last instantiated.

5.2.3.2 SNMP Group

This group is a collection of objects providing basic instrumentation and control of an SNMP entity.

Some objects with a Status of “obsolete” are supported. This is to comply with a request from Technical Support.

OIDs:

- snmp - { mib-2 11 } 1.3.6.1.2.1.1.11

Table 8. SNMP Group (1 of 2)

Object	OID	Syntax	Access	Status	Supported
snmpInPkts	{ snmp 1}	Couner32	read-only	current	Y
snmpOutPkts	{ snmp 2}	Couner32	read-only	obsolete	N
snmpInBadVersions	{ snmp 3}	Couner32	read-only	current	Y
snmpInBadCommunityNames	{ snmp 4}	Couner32	read-only	current	Y
snmpInBadCommunityUses	{ snmp 5 }	Couner32	read-only	current	Y
snmpInASNParseErrs	{ snmp 6 }	Couner32	read-only	current	Y
	{ snmp 7 }[a]				
snmpInTooBigs	{ snmp 8 }	Counter	read-only	obsolete	N
snmpInNoSuchNames	{ snmp 9 }	Counter	read-only	obsolete	N
snmpInBadValues	{ snmp 10 }	Counter	read-only	obsolete	N
snmpInReadOnlys	{ snmp 11 }	Counter	read-only	obsolete	N
snmpInGenErrs	{ snmp 12 }	Counter	read-only	obsolete	N
snmpInTotalReqVars	{ snmp 13 }	Counter	read-only	obsolete	N
snmpInTotalSetVars	{ snmp 14 }	Counter	read-only	obsolete	N
snmpInGetRequests	{ snmp 15 }	Counter	read-only	obsolete	Y
snmpInGetNexts	{ snmp 16 }	Counter	read-only	obsolete	Y
snmpInSetRequests	{ snmp 17 }	Counter	read-only	obsolete	Y
snmpInGetResponses	{ snmp 18 }	Counter	read-only	obsolete	N
snmpInTraps	{ snmp 19 }	Counter	read-only	obsolete	N
snmpOutTooBigs	{ snmp 20 }	Counter	read-only	obsolete	N
snmpOutNoSuchNames	{ snmp 21 }	Counter	read-only	obsolete	N
snmpOutBadValues	{ snmp 22 }	Counter	read-only	obsolete	N
	{ snmp 23 }[a]				
snmpOutGenErrs	{ snmp 24 }	Counter	read-only	obsolete	N
snmpOutGetRequests	{ snmp 25 }	Counter	read-only	obsolete	N
snmpOutGetNexts	{ snmp 26 }	Counter	read-only	obsolete	N
snmpOutSetRequests	{ snmp 27 }	Counter	read-only	obsolete	N
snmpOutGetResponses	{ snmp 28 }	Counter	read-only	obsolete	Y
snmpOutTraps	{ snmp 29 }	Counter	read-only	obsolete	Y

Table 8. SNMP Group (2 of 2)

Object	OID	Syntax	Access	Status	Supported
snnpEnableAuthenTraps	{ snmp 30 }	INTEGER enabled(1) disabled(2)	read-write	current	Y
snmpSilentDrops	{ snmp 31 }	Couner32	read-only	current	Y
snmpProxyDrops	{ snmp 31 }	Couner32	read-only	current	Y

a. This OID is not used.

5.2.3.2.1 snmpInPkts

The total number of messages delivered to the SNMP entity from the transport service.

5.2.3.2.2 snmpInBadVersions

The total number of SNMP messages which were delivered to the SNMP entity and were for an unsupported SNMP version.

5.2.3.2.3 snmpInBadCommunityNames

The total number of SNMP messages delivered to the SNMP entity which used a SNMP community name not known to said entity.

5.2.3.2.4 snmpInBadCommunityUses

The total number of SNMP messages delivered to the SNMP entity which represented an SNMP operation which was not allowed by the SNMP community named in the message.

5.2.3.2.5 snmpInASNParseErrs

The total number of ASN.1 or BER errors encountered by the SNMP entity when decoding received SNMP messages.

5.2.3.2.6 snmpEnableAuthenTraps

Indicates whether the SNMP entity is permitted to generate authenticationFailure traps. The value of this object overrides any configuration information; as such, it provides a means whereby all authenticationFailure traps may be disabled.

Note that it is strongly recommended that this object be stored in non-volatile memory so that it remains constant across re-initializations of the network management system.

This object can take on the following values:

- enabled(1)
- disabled(2)

5.2.3.2.7 snmpSilentDrops

The total number of GetRequest-PDUs, GetNextRequest-PDUs, GetBulkRequest-PDUs, SetRequest-PDUs, and InformRequest-PDUs delivered to the SNMP entity which were silently dropped because the size of a reply containing an alternate Response-PDU with an empty variable-bindings field was greater than either a local constraint or the maximum message size associated with the originator of the request.

5.2.3.2.8 snmpProxyDrops

The total number of GetRequest-PDUs, GetNextRequest-PDUs, GetBulkRequest-PDUs, SetRequest-PDUs, and InformRequest-PDUs delivered to the SNMP entity which were silently dropped because the transmission of the (possibly translated) message to a proxy target failed in a manner (other than a time-out) such that no Response-PDU could be returned.

5.2.3.3 Set Group

This group is a collection of objects which allow several cooperating SNMPv2 entities, all acting in a manager role, to coordinate their use of the SNMPv2 set operation.

OIDs:

- snmpSet - { snmpMIBObjects 6 } 1.3.6.1.6.3.1.1.6

Table 9. Set Group

Object	OID	Syntax	Access	Status	Supported
snmpSetSerialNo	{ snmpSet 1}	TestAndIncr	read-write	current	N

5.2.3.3.1 snmpSetSerialNo

An advisory lock used to allow several cooperating SNMPv2 entities, all acting in a manager role, to coordinate their use of the SNMPv2 set operation.

This object is used for coarse-grain coordination. To achieve fine-grain coordination, one or more similar objects might be defined within each MIB group, as appropriate.

5.2.4 Traps

This section defines a collection of objects which allow the SNMPv2 entity, when acting in an agent role, to be configured to generate SNMPv2-Trap-PDUs. Note that the BitStorm 2600 and GranDSLAM 4200 IP DSLAM send all traps as SNMPv1 PDUs (reference Traps, Section 2, Page 3).

The **snmpTrapOID** is the authoritative identification of the notification currently being sent. This variable occurs as the second varbind in every SNMPv2-Trap-PDU and InformRequest-PDU.

The **snmpTrapEnterprise** is the authoritative identification of the enterprise associated with the trap currently being sent. When a SNMPv2 proxy agent is mapping an RFC 1157 Trap-PDU into a SNMPv2-Trap-PDU, this variable occurs as the last varbind.

OIDs:

- snmpTraps - { snmpMIBObjects 5 } 1.3.6.1.6.3.1.1.5

Table 10. SNMPv2-MIB Traps

Notification	OID	Variable Bindings / Objects[a]	Status	Supported
coldStart	{ snmpTraps 1}		current	N
warmStart	{ snmpTraps 2}		current	Y
authenticationFailure	{ snmpTraps 5}	devAuthenticationFailureIpAddress	current	Y

a. The snmpTrapOID and snmpTrapEnterprise objects apply only to SNMPv2-Trap-PDUs and InformRequest-PDUs and will not be supported in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM as it those PDU types are not used to generate trap/notifications.

5.2.4.1 coldStart

A coldStart trap signifies that the SNMPv2 entity, acting in an agent role, is reinitializing itself and that its configuration may have been altered.

5.2.4.2 warmStart

A warmStart trap signifies that the SNMPv2 entity, acting in an agent role, is reinitializing itself such that its configuration is unaltered.

5.2.4.3 authenticationFailure

An authenticationFailure trap signifies that the SNMPv2 entity, acting in an agent role, has received a protocol message that is not properly authenticated. While all implementations of the SNMPv2 must be capable of generating this trap, the snmpEnableAuthenTraps object indicates whether this trap is generated.

5.2.4.3.1 devAuthenticationFailureIpAddress

The following is an excerpt from the PDN-HEALTHANDSTATUS-MIB, pdn_HealthAndStatus.mib:

> Authentication Failure Ip Address is used when a authenticationFailure trap is generated. The single varbind represents the ip address causing the error. If the ip address is not known, 0.0.0.0 is used.
>
> OID:
> — devAuthenticationFailureIpAddress-{ devStatus 5 }1.3.6.1.4.1.1795.2.24.2.4.1.5
>
> Syntax:
> — IpAddress
>
> Access:
> — read-only
>
> Status:
> — mandatory

5.3 IF-MIB — RFC 2863

5.3.1 Introduction

This MIB describes objects for managing interfaces. The RFC used as a reference was dated June 2000. The LAST-UPDATED for the IF-MIB was “200006140000Z.”

Figure , Interface Diagram, page 31, depicts the interfaces excluding those that exist on the Management and Uplink Interface Module (MAUI). Figure , GigE MAUI, page 32, depicts the interfaces that exist on the GigE MAUI. Figure , T1/E1 ML PPP MAUI, page 33 depicts the interfaces that exist on the T1/E1 ML PPP MAUI. In these figures, there are three-digit number after the port name. These three-digit numbers represent the Port/Interface Number for the particular port from Table 15, ifIndex, page 39. Table 11, Interface List, page 29 lists all the interfaces.

The T1/E1 ML PPP Uplink MAUI, Figure , page 33, supports up to 8 multilink PPP bundles. However, typically there will be multiple T1/E1 port assigned to a multilink PPP bundle and multiple bundles are supported.

Table 11. Interface List (1 of 2)

Interface	Quantity	Figure Label
Console Port	1 per Chassis	Console
Ethernet Port 1 (10/100 Base T)	1 per Chassis	Eth1
Ethernet Port 2 (10/100 Base T)	1 per Chassis	Eth2
Ethernet Port 3 (1000 Base T) • 1000 Base T • GigE Small Form-factor Pluggable	1 per Chassis[a]	Eth3
T1/E1 Ports[b]	8 per Chassis	T1/E1
PPP HDLC Interface	8 per Chassis	HDLC
PPP LCP Interface	8 per Chassis	LCP
ML PPP Bundle Interfaces	8 per Chassis	ML PPP Bundle
PPP NCP Interface	8 per Chassis	NCP
DSL Ports • ADSL • ADSL2 • ADSL2+ • ReachDSL2.2 • SHDSL.bis	24 per Chassis[c] 48 per Chassis[d]	DSL
DSL ATM Interfaces	24 per Chassis[c] 48 per Chassis[d]	DSL ATM
DSL Ethernet Interfaces	24 per Chassis[c] 48 per Chassis[d]	DSL Eth
Management In-Band Ethernet Interface	1 per Chassis	Mgmt IB Eth

Table 11. Interface List (2 of 2)

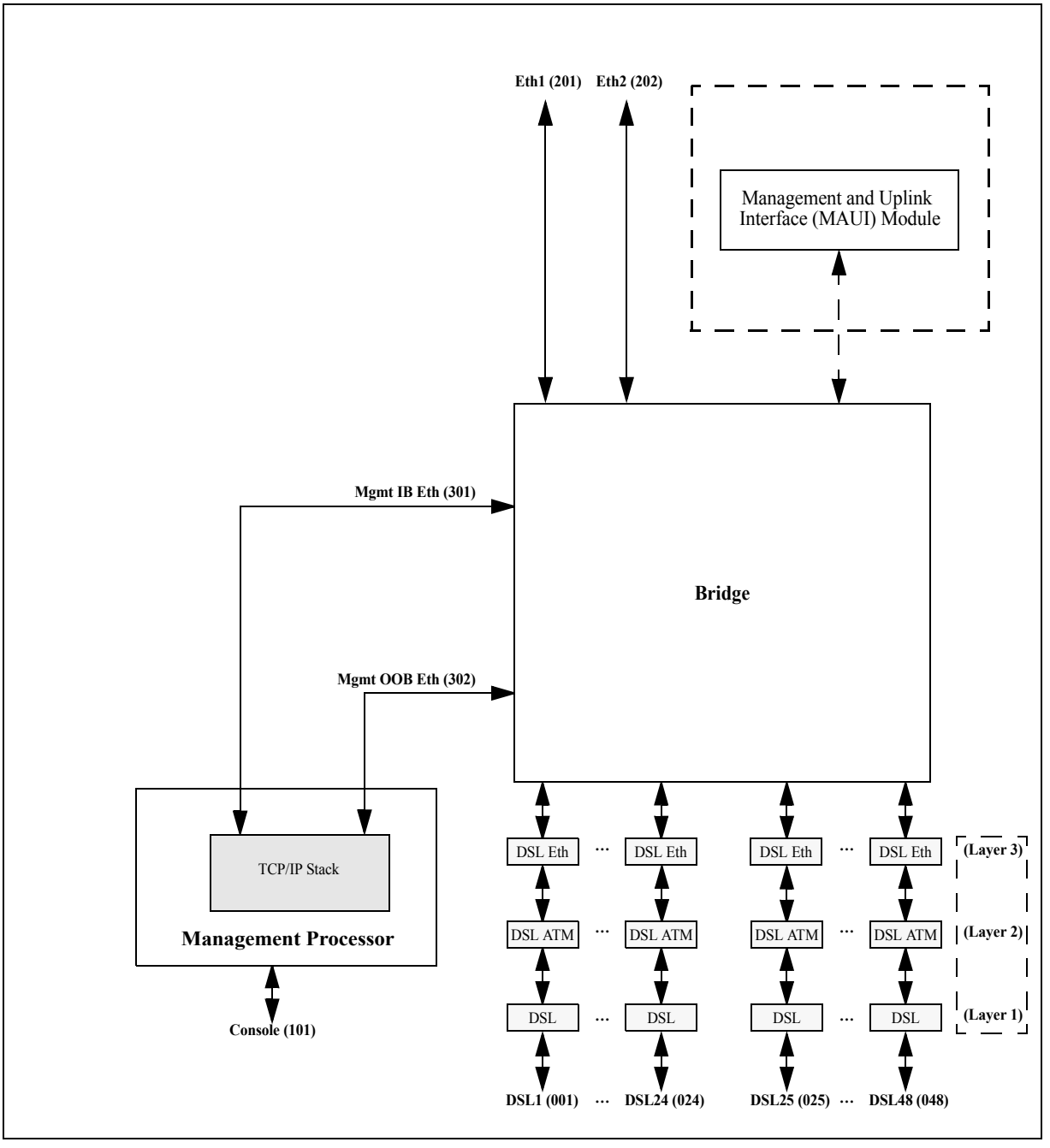
Interface	Quantity	Figure Label
Management Out-of-Band Ethernet Interface	1 per Chassis	Mgmt OOB Eth

a. Not available on units with T1/E1 ML PPP MAUI

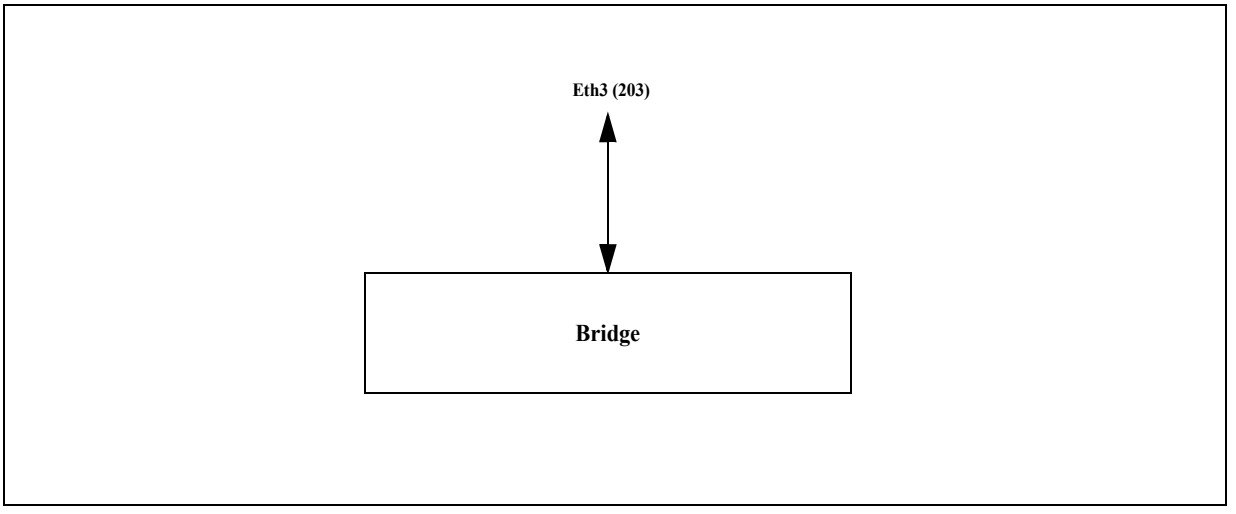
b. Not available on units with GigE MAUI

c. 24-Port Main DSL Card

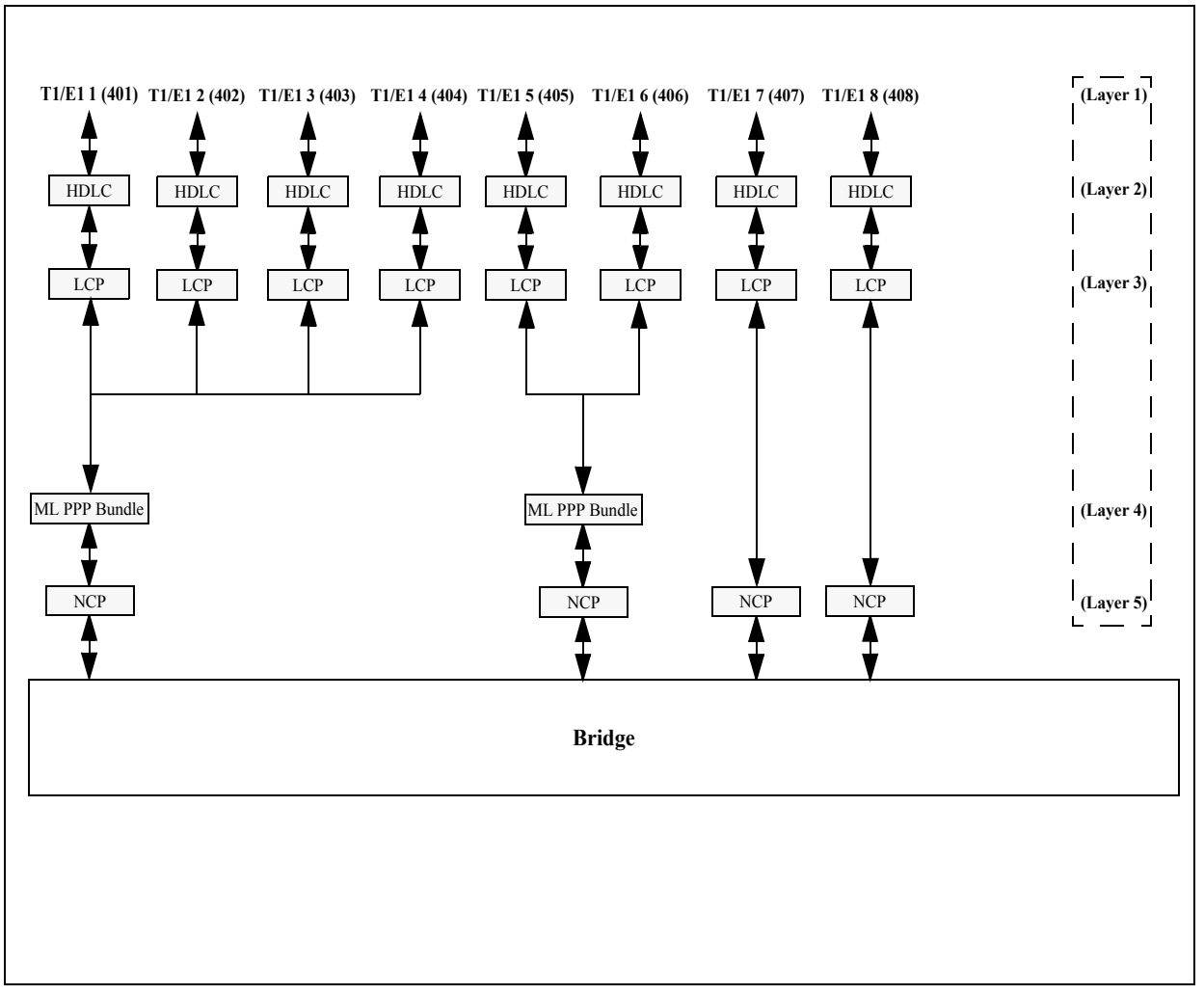
d. 24-Port Main DSL Card plus 24-Port Child DSL Card



Interface Diagram



GigE MAUI



T1/E1 ML PPP MAUI

5.3.2 Textual Conventions

- OwnerString - This data type is used to model an administratively assigned name of the owner of a resource. This information is taken from the NVT ASCII character set. It is suggested that this name contain one or more of the following: ASCII form of the manager station's transport address, management station name (for example, domain name), network management personnel's name, location, or phone number. In some cases the agent itself is the owner of an entry. In these cases, this string shall be set to a string starting with “agent.”

  OwnerString has the same semantics as used in RFC 1271

  The syntax for this type is OCTET STRING (SIZE(0…255))

- InterfaceIndex - A unique value, greater than zero, for each interface or interface sub-layer in the managed system. It is recommended that values are assigned contiguously starting from 1. The value for each interface sub-layer must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.
  InterfaceIndex contains the semantics of ifIndex and should be used for any objects defined in other MIB modules that need these semantics.

  The syntax for this type is integer32 (1…2147483647)

- InterfaceIndexOrZero - This textual convention is an extension of the InterfaceIndex convention. The latter defines a greater than zero value used to identify an interface or interface sub-layer in the managed system. This extension permits the additional value of zero. the value zero is object-specific and must therefore be defined as part of the description of any object which uses this syntax. Examples of the usage of zero might include situations where interface was unknown, or when none or all interfaces need to be referenced.

  The syntax for this object is integer32 (0…2147483647).

5.3.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- interfaces - { mib-2 2 } 1.3.6.1.2.1.2
- ifMIB - { mib-2 31 } 1.3.6.1.2.1.31
- ifMIBObjects - { ifMIB 1 } 1.3.6.1.2.1.31.1

- snmpV2 - { internet 6 } 1.3.6.1.6
- snmpModules - { snmpV2 3 } 1.3.6.1.6.3
- snmpMIB - { snmpModules 1 } 1.3.6.1.6.3.1
- snmpMIBObjects - { snmpMIB 1 } 1.3.6.1.6.3.1.1
- snmpTraps - { snmpMIBObjects 5 } 1.3.6.1.6.3.1.1.5

5.3.4 MIB Objects

The table below identifies the objects called out in this MIB. Any table object that has elements that are supported are indicated by a “Y” in the Supported column.

Table 12. IF-MIB Objects

Object	OID	Syntax	Access	Status	Supported
ifNumber	{ interfaces 1 }	Integer32	read-only	current	Y
ifTable	{ interfaces 3 }	Sequence of ifEntry	not-accessible	current	Y
ifXTable	{ ifMIBObjects 1 }	Sequence of ifXEntry	not-accessible	current	Y
ifStackTable	{ ifMIBObjects 2 }	Sequence of ifStackEntry	not-accessible	current	Y
ifTestTable	{ ifMIBObjects 3 }	Sequence of ifTestEntry	not-accessible	deprecated	N
ifRcvAddressTable	{ ifMIBObjects 4 }	Sequence of ifRcvAddressTable	not-accessible	current	Y
ifTableLastChange	{ ifMIBObjects 5 }	TimerTicks	read-only	current	Y
ifStackLastChange	{ ifMIBObjects 6 }	TimerTicks	read-only	current	Y

5.3.4.1 ifNumber

The number of network interfaces (regardless of their current state) present on this system.

Valid Object Value(s)

The ifNumber will represent the number of rows in the ifTable.

5.3.4.2 ifTableLastChange

The value of sysUpTime at the time of the last creation or deletion of an entry in the ifTable. If the number of entries has been unchanged since the last re-initialization of the local network management subsystem, then this object contains a zero value.

5.3.4.3 ifTable

The Interfaces table contains information on the entity's interfaces. Each sub-layer below the internetwork-layer of a network interface is considered to be an interface. This object type is a sequence of the ifEntry where an entry containing management information applicable to a particular interface.

Index:

- ifIndex

OIDs:

- ifTable - { interfaces 2 } 1.3.6.1.2.1.2.2
- ifEntry - { ifTable 1 } 1.3.6.1.2.1.2.2.1

Table 13. ifTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
ifIndex	{ ifEntry 1 }	Interface Index	read-only	current	Y
ifDescr	{ ifEntry 2 }	DisplayString	read-only	current	Y
ifType	{ ifEntry 3 }	IANAifType	read-only	current	Y
ifMtu	{ ifEntry 4 }	Integer32	read-only	current	Y
ifSpeed	{ ifEntry 5 }	Gauge32	read-only	current	Y
ifPhysAddress	{ ifEntry 6 }	PhysAddress	read-only	current	Y
ifAdminStatus	{ ifEntry 7 }	INTEGER up(**1**) down(**2**) testing(**3**)	read-write	current	Y
ifOperStatus	{ ifEntry 8 }	INTEGER up(**1**) down(**2**) testing(**3** unknown(**4**) dormant(**5**) notPresent(**6**) lowerLayerDown(**7**)	read-only	current	Y
ifLastChange	{ ifEntry 9 }	TimerTicks	read-only	current	Y
ifInOctets	{ ifEntry 10 }	Counter32	read-only	current	Y
ifInUcastPkts	{ ifEntry 11 }	Counter32	read-only	current	Y
ifInNUcastPkts	{ ifEntry 12 }	Counter32	read-only	deprecated	N
ifInDiscards	{ ifEntry 13 }	Counter32	read-only	current	Y
ifInErrors	{ ifEntry 14 }	Counter32	read-only	current	Y
ifInUnknownProtos	{ ifEntry 15 }	Counter32	read-only	current	Y
ifOutOctets	{ ifEntry 16 }	Counter32	read-only	current	Y
ifOutUcastPkts	{ ifEntry 17 }	Counter32	read-only	current	Y

Table 13. ifTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
ifOutNUcastPkts	{ ifEntry 18 }	Counter32	read-only	deprecated	N
ifOutDiscards	{ ifEntry 19 }	Counter32	read-only	current	Y
ifOutErrors	{ ifEntry 20 }	Counter32	read-only	current	Y
ifOutQLen	{ ifEntry 21 }	Counter32	read-only	deprecated	N
ifSpecific	{ ifEntry 22 }	Counter32	read-only	deprecated	N

5.3.4.3.1 ifIndex

A unique value, greater than zero, for each interface. It is recommended that values are assigned contiguously starting from 1. The value for each interface sub-layer must remain constant at least from one re-initialization of the entity's network management system to the next reinitialization.

Valid Object Value(s)

This ifIndex scheme was developed by a committee to solve a general stack hierarchy shown in Figure .

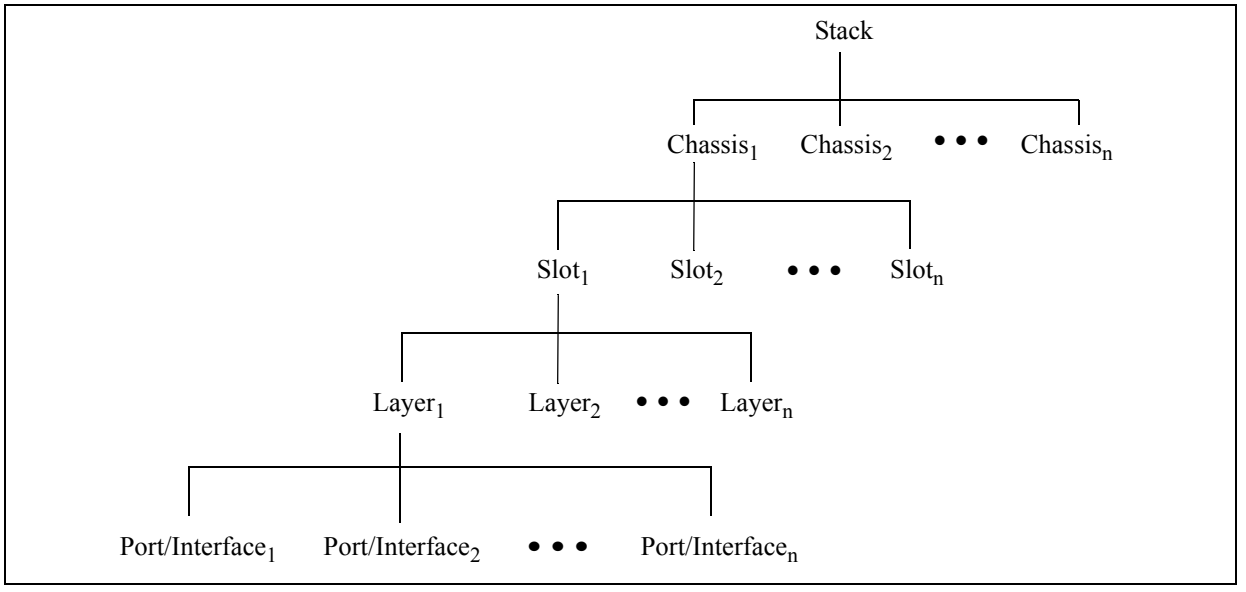
The ifIndex is encoded as a decimal number in the form of:

Where:

- R - Reserved bits. Implied as 0.
- CC - Chassis Number with the range of 1…99 and 0 is a special case to denote N/A. The following are term used to refer to a chassis: chassis, slice, unit, or box.
- SS - Slot Number with the range of 1…99 and 0 is a special case to denote N/A. Slot Number is not applicable to this product and will always be 00.
- L - Layer Number in the range of 1…9 and 0 is a special case to denote N/A.
- PPP - Port/Interface Number in the range of 1…999 and 0 is a special case to denote N/A.

The layer number, L, can be seen as it is meant to be applied in Figure .

Note: Leading zeros may be omitted when using the ifIndex.



Stack Hierarchy

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM products, although considered “stackable,” are not managed in terms of a Single Managed Entity.

Table 14. Interface Stack

Layer	Console	T1/E1 ML PPP	DSL[a]	Eth_n	Mgmt Eth_n
5		NCP			
4		ML PPP Bundle			
3		LCP	DSL Eth		
2		HDLC	DSL ATM		
1	RS232	T1/E1	DSL	Ethernet	Ethernet

a. ADSL, ReachDSL, and SHDSL

Table 15. ifIndex

Interface	Value[a]										Comment
	R	R	C[b]	C[b]	S[c]	S[c]	L	P	P	P	
DSL Ports	0	0	0	1	0	0	1	a	a	a	001…048
DSL ATM Interfaces	0	0	0	1	0	0	2	a	a	a	001…048
DSL Ethernet Interfaces	0	0	0	1	0	0	3	a	a	a	001…048
Console Port	0	0	0	1	0	0	1	1	0	1	
Ethernet Port 1 (10/100 Base T)	0	0	0	1	0	0	1	2	0	1	
Ethernet Port 2 (10/100 Base T)	0	0	0	1	0	0	1	2	0	2	
Ethernet Port 3 (1000 Base T / SFP)	0	0	0	1	0	0	1	2	0	3	
Management In-Band Ethernet Interface	0	0	0	1	0	0	1	3	0	1	
Management Out-of-Band Ethernet Interface	0	0	0	1	0	0	1	3	0	2	
T1/E1 Ports	0	0	0	1	0	0	1	a	a	a	401…408
PPP HDLC Interface	0	0	0	1	0	0	2	a	a	a	401…408
PPP LCP Interface	0	0	0	1	0	0	3	a	a	a	401…408
ML PPP Bundle Interfaces	0	0	0	1	0	0	4	a	a	a	401…408
PPP NCP Interface	0	0	0	1	0	0	5	a	a	a	401…408

a. L PPP of 1 900 to 1 999 is reserved for internal use.

b. denotes the chassis/slice/unit/box number which is N/A in this product.

c. denotes the slot number which is N/A in this product.

Note: Leading zeros may be omitted.

5.3.4.3.2 ifDescr

A textual string containing information about the interface. This string should include the name of the manufacturer, the product name and the version of the interface hardware/software.

Valid Object Value(s)

Table 16. ifDescr (1 of 3)

Interface	Value
DSL Ports	**“DSL PORT 1"** **“DSL PORT 2"** • • • **“DSL PORT 48"**

Table 16. ifDescr (2 of 3)

Interface	Value
DSL ATM Interfaces	**“DSL ATM Interface 1"** **“DSL ATM Interface 2"** • • • **“DSL ATM Interface 48"**
DSL Ethernet Interfaces	**“DSL Ethernet Interface 1"** **“DSL Ethernet Interface 2"** • • • **“DSL Ethernet Interface 48"**
Console Port	**“CONSOLE PORT”**
Ethernet Port 1 (10/100 Base T)	**“ETHERNET PORT 1”**
Ethernet Port 2 (10/100 Base T)	**“ETHERNET PORT 2”**
Ethernet Port 3 (1000 Base T / SFP)	**“ETHERNET PORT 3”**
Management In-Band Ethernet Interface	**“Management In-Band Ethernet Interface”**
Management Out-of-Band Ethernet Interface	**“Management Out-of-Band Ethernet Interface”**
T1/E1 Ports	**“T1/E1 Port 1"** **“T1/E1 Port 2"** • • • **“T1/E1 Port 8"**
PPP HDLC Interface	**“PPP HDLC Interface 1"** **“PPP HDLC Interface 2"** • • • **“PPP HDLC Interface 8"**
PPP LCP Interface	**“PPP LCP Interface 1"** **“PPP LCP Interface 2"** • • • **“PPP LCP Interface 8"**

Table 16. ifDescr (3 of 3)

Interface	Value
ML PPP Bundle Interfaces	**"ML PPP Bundle Interface 1"** **"ML PPP Bundle Interface 2"** • • • **"ML PPP Bundle Interface 8"**
PPP NCP Bridge Interface	**"PPP NCP Interface 1"** **"PPP NCP Interface 2"** • • • **"PPP NCP Interface 8"**

5.3.4.3.3 ifType

The type of interface. Additional values for ifType are assigned by the Internet Assigned Numbers Authority (IANA), through updating the syntax of the IANAifType textual convention. The IANAifType is documented in the IANAifType-MIB found at http://www.iana.org/assignments/ianaiftype-mib.

Valid Object Value(s)

Table 17. ifType

Interface	Value[a,b]
DSL Ports	**adsl(94)** **shdsl(169)**
DSL ATM Interfaces	**atmVirtual(149)**
DSL Ethernet Interfaces	**ethernetCsmacd(6)**
Console Port	**rs232(33)**
Ethernet Port 1(10/100 Base T)	**ethernetCsmacd(6)**
Ethernet Port 2 (10/100 Base T)	**ethernetCsmacd(6)**
Ethernet Port 3 (1000 Base T / SFP)	**ethernetCsmacd(6)**
Management In-Band Ethernet Interface	**ethernetCsmacd(6)**
Management Out-of-Band Ethernet Interface	**ethernetCsmacd(6)**
T1/E1 Ports	**ds1(18)**[c]
PPP HDLC Interface	**hdlc(118)**
PPP LCP Interface	**ppp(23)**
ML PPP Bundle Interfaces	**pppMultilinkBundle(108)**
PPP NCP Interface	**ppp(23)**

a. RFC 2556, “Ethernet-Like MIB,” August 1999, recommends in section 3.2.6 ifType, all Ethernet-like interfaces use an ifType of ethernetCsmacd(6). The particular type of Ethernet that the interface is running is available from ifSpeed and the ifMauType in the 802.3 MAU MIB (RFC 2668).

b. Both ReachDSL and ADSL Ports will report ifType adsl(94).

c. e1(19) is obsolete as noted in RFC 2495, DS1-MIB.

5.3.4.3.4 ifMtu

The size of the largest packet which 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.

Valid Object Value(s)

Table 18. ifMtu

Interface	Value
DSL Ports	**53**
DSL ATM Interfaces	**53**
DSL Ethernet Interfaces	**1550**
Console Port	**0**
Ethernet Port 1 (10/100 Base T)	**1550**
Ethernet Port 2 (10/100 Base T)	**1550**
Ethernet Port 3 (1000 Base T / SFP)	**1550**
Management In-Band Ethernet Interface	**1550**
Management Out-of-Band Ethernet Interface	**1550**
T1/E1 Ports	**0**
PPP HDLC Interface	**1532**
PPP LCP Interface	**1530**
ML PPP Bundle Interfaces	**1528**
PPP NCP Interface	**1528**

5.3.4.3.5 ifSpeed

An estimate of the interface's current bandwidth in bits per second. For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth. If the bandwidth of the interface is greater than the maximum value reportable by this object then this object should report its maximum value (4,294,967,295) and ifHighSpeed must be used to report the interlaces speed. For a sub-layer which has no concept of bandwidth, this object should be zero.

Valid Object Value(s)

Note that in the table below, spaces are added to enhance readability.
For example, 10M bps is displayed as 10000000 but documented in the table below as 10 000 000.

Table 19. ifSpeed (1 of 2)

Interface	Value
DSL Ports	• Port Trained - **The downstream rate of the port** • Port Not Trained - **Zero**
DSL ATM Interfaces	• Port Trained - **The downstream rate of the port** • Port Not Trained - **Zero**
DSL Ethernet Interfaces	**100 000 000**
Console Port	**Current configured speed** The possible values are: 9600 19200, 38400, 57600, and 115200 bps
Ethernet Port 1 (10/100 Base T)	The possible values are: **10 000 000** or **100 000 000** • Auto-negotiation disabled — currently optioned speed • Auto-negotiation enabled and completed — auto-negotiation speed • Auto-negotiation enabled and not completed — **100 000 000**
Ethernet Port 2 (10/100 Base T)	The possible values are: **10 000 000** or **100 000 000** • Auto-negotiation disabled — currently optioned speed • Auto-negotiation enabled and completed — auto-negotiation speed • Auto-negotiation enabled and not completed — **100 000 000**
Ethernet Port 3 (1000 Base T / SFP)	The possible values are: **10 000 000**, **100 000 000**, or **1 000 000 000** • Auto-negotiation disabled — currently poisoned speed • Auto-negotiation enabled and completed — auto-negotiation speed • Auto-negotiation enabled and not completed — **1 000 000 000** When running on the SFP, the only valid value is **1 000 000 000**
Management In-Band Ethernet Interface	**100 000 000**
Management Out-of-Band Ethernet Interface	**100 000 000**

Table 19. ifSpeed (2 of 2)

Interface	Value
T1/E1 Ports	**Current configured speed**[a] • 1 544 000 if configured for T1 • 2 048 000 if configured for E1 • 0 if non of the above
PPP HDLC Interface	**Current configured speed** The possible values are: 64 000 bps to 2 048 000 bps in 64 000 bps increments
PPP LCP Interface	**0**
ML PPP Bundle Interfaces	**The sum of the ifSpeed for each T1/E1 port that is currently part of the ML PPP Bundle.**
PPP NCP Interface	**0**

a. The definition of being configured for T1 or E1 is when the appropriate dsx1LineType, Section 6.14.4.1.5, Page 351 and dsx1LineCoding, Section 6.14.4.1.6, Page 352 combination has been set.

5.3.4.3.6 ifPhysAddress

The interface's address at its protocol sub-layer. For example, for an 802.x interface, this object normally contains a MAC address. The interface's media-specific MIB must define the bit and byte ordering and the format of the value of this object. For interfaces which do not have such an address (for example, a serial line), this object should contain an octet string of zero length.

Valid Object Value(s)

Table 20. ifPhysAddress (1 of 2)

Interface	Value
DSL Ports	**An octet string of zero length**
DSL ATM Interfaces	**An octet string of zero length**
DSL Ethernet Interfaces	**An octet string of zero length**
Console Port	**An octet string of zero length**
Ethernet Port 1 (10/100 Base T)	**An octet string of zero length**
Ethernet Port 2 (10/100 Base T)	**An octet string of zero length**
Ethernet Port 3 (1000 Base T / SFP)	**An octet string of zero length**
Management In-Band Ethernet Interface	**MAC Address**
Management Out-of-Band Ethernet Interface	**MAC Address**
T1/E1 Ports	**An octet string of zero length**
PPP HDLC Interface	**An octet string of zero length**
PPP LCP Interface	**An octet string of zero length**

Table 20. ifPhysAddress (2 of 2)

Interface	Value
ML PPP Bundle Interfaces	**An octet string of zero length**
PPP NCP Interface	**An octet string of zero length**

5.3.4.3.7 ifAdminStatus

The desired state of the interface.

- up(1) - The interface is operational and operational packets can be passed
- down(2) - When a managed system is initialized, all interfaces start with ifAdminStatus in the down(2) state
- testing(3) - The interface is in some test state and no operational packets can be passed

Valid Object Value(s)

Table 21. ifAdminStatus (1 of 2)

Interface	Value
DSL Ports	Any value supported by the syntax
DSL ATM Interfaces	The only valid value is **up(1)** Setting the object to any other value will result in **badValue**
DSL Ethernet Interfaces	The only valid value is **up(1)** Setting the object to any other value will result in **badValue**
Console Port	The only valid value is **up(1)** Setting the object to any other value will result in **badValue**
Ethernet Port 1 (10/100 Base T)	The only valid values are **up(1)** or **down(2)** Setting the object to any other value will result in **badValue**
Ethernet Port 2 (10/100 Base T)	The only valid values are **up(1)** or **down(2)** Setting the object to any other value will result in **badValue**
Ethernet Port 3 (1000 Base T / SFP)	The only valid values are **up(1)** or **down(2)** Setting the object to any other value will result in **badValue**
Management In-Band Ethernet Interface	The only valid value is **up(1)** Setting the object to any other value will result in **badValue**
Management Out-of-Band Ethernet Interface	The only valid value is **up(1)** Setting the object to any other value will result in **badValue**
T1/E1 Ports	Any value supported by the syntax

Table 21. ifAdminStatus (2 of 2)

Interface	Value
PPP HDLC Interface	Any value supported by the syntax
PPP LCP Interface	The only valid values are **up(1)** or **down(2)** Setting the object to any other value will result in **badValue**
ML PPP Bundle Interfaces	Any value supported by the syntax
PPP NCP Interface	Any value supported by the syntax

5.3.4.3.8 ifOperStatus

The current operational state of the interface.

- up(1) - Ready to pass packets
- down(2) -
- testing(3) - The interface is in some test state and no operational packets can be passed.
- unknown(4) - status can not be determined for some reason.
- dormant(5), -
- notPresent(6) - some component is missing
- lowerLayerDown(7) - down due to state of lower-layer interface(s)

The testing(3) state indicates that no operational packets can be passed. If ifAdminStatus is down(2) then ifOperStatus should be down(2). If ifAdminStatus is changed to up(1) then ifOperStatus should change to up(1) if the interface is ready to transmit and receive network traffic; it should change to dormant(5) if the interface is waiting for external actions (such as a serial line waiting for an incoming connection); it should remain in the down(2) state if and only if there is a fault that prevents it from going to the up(1) state; it should remain in the notPresent(6) state if the interface has missing (typically, hardware) components.

Valid Object Value(s)

Table 22. ifOperStatus (1 of 2)

Interface	Value
DSL Ports	Any of the values supported by the syntax
DSL ATM Interfaces	Any of the values supported by the syntax except **down(2)**
DSL Ethernet Interfaces	Any of the values supported by the syntax except **down(2)**
Console Port	The port is always **up(1)**
Ethernet Port 1 (10/100 Base T)	Any value supported by the syntax
Ethernet Port 2 (10/100 Base T)	Any value supported by the syntax
Ethernet Port 3 (1000 Base T / SFP)	Any value supported by the syntax
Management In-Band Ethernet Interface	Any of the values supported by the syntax except **down(2)**
Management Out-of-Band Ethernet Interface	Any of the values supported by the syntax except **down(2)**
T1/E1 Ports	Any of the values supported by the syntax
PPP HDLC Interface	Any of the values supported by the syntax

Table 22. ifOperStatus (2 of 2)

Interface	Value
PPP LCP Interface	Any of the values supported by the syntax
ML PPP Bundle Interfaces	Any value supported by the syntax
PPP NCP Interface	Any value supported by the syntax

5.3.4.3.9 ifLastChange

The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last re-initialization of the local network management subsystem, then this object contains a zero value.

5.3.4.3.10 Input Counters

Table 23. ifIn Counter Objects (1 of 2)

Interface		ifInOctets	ifInUcastPkts	ifInNUcastPkts[a]	ifInDiscards	ifInErrors	ifInUnknownProtos
DSL Ports	ADSL	**S**	U	U	U	**S**	U
	ReachDSL	**S**	U	U	**S**	**S**	U
	SHDSL	**S**	U	U	U	**S**	U
DSL ATM Interfaces	ADSL	**S**	U	U	U	**S**	U
	ReachDSL	**S**	U	U	U	**S**	**S**
	SHDSL	**S**	U	U	U	**S**	U
DSL Ethernet Interfaces		U	**S**	U	**S**	**S**	**S**
Console Port		U	U	U	U	U	U
Ethernet Port 1 (10/100 Base T)		**S**	**S**	U	**S**	**S**	U
Ethernet Port 2 (10/100 Base T)		**S**	**S**	U	**S**	**S**	U
Ethernet Port 3 (1000 Base T / SFP)		**S**	**S**	U	**S**	**S**	U
Management In-Band Ethernet Interface		**S**	**S**	U	U	U	**S**
Management Out-of-Band Ethernet Interface		**S**	**S**	U	U	U	**S**
T1/E1 Ports		U	U	U	U	U	U
PPP HDLC Interface		U	U	U	U	U	U
PPP LCP Interface		U	U	U	U	U	U
ML PPP Bundle Interfaces		U	U	U	U	U	U
PPP NCP Interface		U	U	U	U	U	U

Table 23. ifIn Counter Objects (2 of 2)

Interface	ifInOctets	ifInUcastPkts	ifInNUcastPkts[a]	ifInDiscards	ifInErrors	ifInUnknownProtos
Legend:S = Supported U = Unsupported (returns “No Such Name”) Z = Unsupported (returns zero)						

a. This MIB object is deprecated.

5.3.4.3.10.1 ifInOctets

The total number of octets received on the interface, including framing characters.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.10.2 ifInUcastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.10.3 ifInNUcastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast or broadcast address at this sub-layer.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

This object is deprecated in favor of ifInMulticastPkts and ifInBroadcastPkts.

5.3.4.3.10.4 ifInDiscards

The number of inbound packets which 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.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.10.5 ifInErrors

For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For character- oriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.10.6 ifInUnknownProtos

For packet-oriented interfaces, the number of packets received via the interface which were discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing the number of transmission units received via the interface which were discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter will always be 0.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.11 Output Counters

Table 24. ifOut Counter Objects

Interface		ifOutOctets	ifOutUcastPkts	ifOutNUcastPkts[a]	ifOutDiscards	ifOutErrors
DSL Ports	ADSL	**S**	U	U	**S**	U
	ReachDSL	**S**	U	U	**S**	U
	SHDSL	**S**	U	U	**S**	U
DSL ATM Interfaces	ADSL	**S**	U	U	**S**	U
	ReachDSL	**S**	U	U	**S**	U
	SHDSL	**S**	U	U	**S**	U
DSL Ethernet Interfaces		U	**S**	U	U	U
Console Port		U	U	U	U	U
Ethernet Port 1 (10/100 Base T)		**S**	**S**	U	**S**	**S**
Ethernet Port 2 (10/100 Base T)		**S**	**S**	U	**S**	**S**
Ethernet Port 3 (1000 Base T / SFP)		**S**	**S**	U	U	U
Management In-Band Ethernet Interface		**S**	**S**	U	U	U
Management Out-of-Band Ethernet Interface		**S**	**S**	U	U	U
T1/E1 Ports		U	U	U	U	U
PPP HDLC Interface		U**S**	U	U	U	U
PPP LCP Interface		U	U	U	U	U
ML PPP Bundle Interfaces		U	U	U	U	U
PPP NCP Interface		U	U	U	U	U

Legend:S = Supported
U = Unsupported (returns “No Such Name”)
Z = Unsupported (returns zero)

a. This MIB object is deprecated.

5.3.4.3.11.1 ifOutOctets

The total number of octets transmitted out of the interface, including framing characters.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.11.2 ifOutUcastPkts

The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.11.3 ifOutNUcastPkts

The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

This object is deprecated in favor of ifOutMulticastPkts and ifOutBroadcastPkts.

5.3.4.3.11.4 ifOutDiscards

The number of outbound packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.11.5 ifOutErrors

For packet-oriented interfaces, the number of outbound packets that could not be transmitted because of errors. For character-oriented or fixed-length interfaces, the number of outbound transmission units that could not be transmitted because of errors.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.3.12 ifOutQLen

The length of the output packet queue (in packets).

Valid Object Value(s)

This object is deprecated and will not be supported.

Valid Object Value(s)

This object is deprecated and will not be supported.

5.3.4.3.13 ifSpecific

A reference to MIB definitions specific to the particular media being used to realize the interface. It is recommended that this value point to an instance of a MIB object in the media-specific MIB, that is, that this object have the semantics associated with the InstancePointer textual convention defined in RFC 2579. In fact, it is recommended that the media-specific MIB specify what value ifSpecific should/can take for values of ifType. If no MIB definitions specific to the particular media are available, the value should be set to the OBJECT IDENTIFIER { 0 0 }.

Valid Object Value(s)

This object is deprecated and will not be supported.

5.3.4.4 ifXTable

The Extension to the interface table provides additional objects for the interface table. The number of entries is given by the value ifNumber. This object type is a sequence of ifXEntry and contains additional management information applicable to a particular interface.

Augments:

- ifTable (Section 5.3.4.3, Page 36)

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- ifXTable - { ifMIBObjects 1 } 1.3.6.1.2.1.31.1.1
- ifXEntry - { ifXTable 1 } 1.3.6.1.2.1.31.1.1.1

Table 25. ifXTable

Object	OID	Syntax	Access	Status	Supported
ifName	{ ifXEntry 1 }	DisplayString	read-only	current	Y
ifInMulticastPkts	{ ifXEntry 2 }	Counter32	read-only	current	Y
ifInBroadcastPkts	{ ifXEntry 3 }	Counter32	read-only	current	Y
ifOutMulticastPkts	{ ifXEntry 4 }	Counter32	read-only	current	Y
ifOutBroadcastPkts	{ ifXEntry 5 }	Counter32	read-only	current	Y
ifHCInOctets	{ ifXEntry 6 }	Counter64	read-only	current	N
ifHCInUcastPkts	{ ifXEntry 7 }	Counter64	read-only	current	N
ifHCInMulticastPkts	{ ifXEntry 8 }	Counter64	read-only	current	N
ifHCInBroadcastPkts	{ ifXEntry 9 }	Counter64	read-only	current	N
ifHCOutOctets	{ ifXEntry 10 }	Counter64	read-only	current	N
ifHCOutUcastPkts	{ ifXEntry 11 }	Counter64	read-only	current	N
ifHCOutMulticastPkts	{ ifXEntry 12 }	Counter64	read-only	current	N
ifHCOutBroadcastPkts	{ ifXEntry 13 }	Counter64	read-only	current	N
ifLinkUpDownTrapEnable	{ ifXEntry 14 }	INTEGER enabled(**1**) disabled(**2**)	read-write	current	Y
ifHighSpeed	{ ifXEntry 15 }	Gauge32	read-only	current	N
ifPromiscuousMode	{ ifXEntry 16 }	TruthValue	read-write	current	Y
IfConnectorPresent	{ ifXEntry 17 }	TruthValue	read-only	current	Y
ifAlias	{ ifXEntry 18 }	DisplayString(1…20)[a]	read-write	current	Y
ifCounterDiscontinuityTime	{ ifXEntry 19 }	TimeStamp	read-only	current	Y

a. Product imposed limit.

5.3.4.4.1 ifName

The textual 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, then each will have the same value of ifName. Note that for an agent which responds to SNMP queries concerning an interface on some other (proxied) device, then the value of ifName for such an interface is the proxied device's local name for it.

If there is no local name, or this object is otherwise not applicable, then this object contains a zero-length string.

Valid Object Value(s)

Table 26. ifName

Interface	Value
DSL Ports	**"dsl *n*"** where n is 1…48
DSL ATM Interfaces	**"dsl atm *n*"** where n is 1…48
DSL Ethernet Interfaces	**"dsl eth *n*"** where n is 1…48
Console Port	**"console"**
Ethernet Port 1 (10/100 Base T)	**"ethernet 1"**
Ethernet Port 2 (10/100 Base T)	**"ethernet 2"**
Ethernet Port 3 (1000 Base T / SFP)	**"ethernet 3"**
Management In-Band Ethernet Interface	**"mgmt ib eth"**
Management Out-of-Band Ethernet Interface	**"mgmt oob eth"**
T1/E1 Ports	**"t1/e1 *n*"** where n is 1…8
PPP HDLC Interface	**"ppp hdlc interface *n*"** where n is 1…8
PPP LCP Interface	**"ppp lcp interface *n*"** where n is 1…8
ML PPP Bundle Interfaces	**"ml ppp bundle interface *n*"** where n is 1…8
PPP NCP Interface	**"ppp ncp interface *n*"** where n is 1…8

5.3.4.4.2 Input Counters

Table 27. ifXIn Counter Objects (1 of 2)

Interface	ifInMulticastPkts	ifInBroadcastPkts
DSL Ports	U	U
DSL ATM Interfaces	U	U

Table 27. ifXIn Counter Objects (2 of 2)

Interface	ifInMulticastPkts	ifInBroadcastPkts
DSL Ethernet Interfaces	U	U
Console Port	U	U
Ethernet Port 1 (10/100 Base T)	**S**	**S**
Ethernet Port 2 (10/100 Base T)	**S**	**S**
Ethernet Port 3 (1000 Base T / SFP)	**S**	**S**
Management In-Band Ethernet Interface	**S**	**S**
Management Out-of-Band Ethernet Interface	**S**	**S**
T1/E1 Ports	U	U
PPP HDLC Interface	U	U
PPP LCP Interface	U	U
ML PPP Bundle Interfaces	U	U
PPP NCP Interface	U	U

Legend:S = Supported
U = Unsupported (returns “No Such Name”)
Z = Unsupported (returns zero)

5.3.4.4.2.1 ifInMulticastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at this sub-layer. For a MAC layer protocol, this includes both Group and Functional addresses.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.2.2 ifInBroadcastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a broadcast address at this sub-layer.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.3 Output Counters

Table 28. ifXOut Counter Objects

Interface	ifOutMulticastPkts	ifOutBroadcastPkts
DSL Ports	U	U
DSL ATM Interfaces	U	U
DSL Ethernet Interfaces	U	U
Console Port	U	U
Ethernet Port 1 (10/100 Base T)	**S**	**S**
Ethernet Port 2 (10/100 Base T)	**S**	**S**
Ethernet Port 3 (1000 Base T / SFP)	**S**	**S**
Management In-Band Ethernet Interface	**S**	**S**
Management Out-of-Band Ethernet Interface	**S**	**S**
T1/E1 Ports	U	U
PPP HDLC Interface	U	U
PPP LCP Interface	U	U
ML PPP Bundle Interfaces	U	U
PPP NCP Interface	U	U

Legend:S = Supported
U = Unsupported (returns “No Such Name”)
Z = Unsupported (returns zero)

5.3.4.4.3.1 ifOutMulticastPkts

The total number of packets that higher-level protocols requested be transmitted, and which 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.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.3.2 ifOutBroadcastPkts

The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a broadcast address at this sub-layer, including those that were discarded or not sent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.4 High Capacity Input Counters

Table 29. ifHCIn Counter Objects

Interface	ifHCInOctets	ifHCInUcastPkts	ifHCInMulticastPkts	ifHCInBroadcastPkts
DSL Ports	U	U	U	U
DSL ATM Interfaces	U	U	U	U
DSL Ethernet Interfaces	U	U	U	U
Console Port	U	U	U	U
Ethernet Port 1 (10/100 Base T)	U	U	U	U
Ethernet Port 2 (10/100 Base T)	U	U	U	U
Ethernet Port 3 (1000 Base T / SFP)	U	U	U	U
Management In-Band Ethernet Interface	U	U	U	U
Management Out-of-Band Ethernet Interface	U	U	U	U
T1/E1 Ports	U	U	U	U
PPP HDLC Interface	U	U	U	U
PPP LCP Interface	U	U	U	U
ML PPP Bundle Interfaces	U	U	U	U
PPP NCP Interface	U	U	U	U

Legend:S = Supported
U = Unsupported (returns “No Such Name”)
Z = Unsupported (returns zero)

5.3.4.4.4.1 ifHCInOctets

The total number of octets received on the interface, including framing characters. This object is a 64-bit version of ifInOctets.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.4.2 ifHCInUcastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were not addressed to a multicast or broadcast address at this sub-layer. This object is a 64-bit version of ifInUcastPkts.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.4.3 ifHCInMulticastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which 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.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.4.4 ifHCInBroadcastPkts

The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a broadcast address at this sub-layer. This object is a 64-bit version of ifInBroadcastPkts.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.5 High Capacity Output Counters

Table 30. ifHCOut Counter Objects

Interface	ifHCOutOctets	ifHCOutUcastPkts	ifHCOutMulticastPkts	ifHCOutBroadcastPkts
DSL Ports	U	U	U	U
DSL ATM Interfaces	U	U	U	U
DSL Ethernet Interfaces	U	U	U	U
Console Port	U	U	U	U
Ethernet Port 1 (10/100 Base T)	U	U	U	U
Ethernet Port 2 (10/100 Base T)	U	U	U	U
Ethernet Port 3 (1000 Base T / SFP)	U	U	U	U
Management In-Band Ethernet Interface	U	U	U	U
Management Out-of-Band Ethernet Interface	U	U	U	U
T1/E1 Ports	U	U	U	U
PPP HDLC Interface	U	U	U	U
PPP LCP Interface	U	U	U	U
ML PPP Bundle Interfaces	U	U	U	U
PPP NCP Interface	U	U	U	U

Legend:S = Supported

U = Unsupported (returns “No Such Name”)

Z = Unsupported (returns zero)

5.3.4.4.5.1 ifHCOutOctets

The total number of octets transmitted out of the interface, including framing characters. This object is a 64-bit version of ifOutOctets.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.5.2 ifHCOutUcastPkts

The total number of packets that higher-level protocols requested be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent. This object is a 64-bit version of ifOutUcastPkts.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.5.3 ifHCOutMulticastPkts

The total number of packets that higher-level protocols requested be transmitted, and which 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.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.5.4 ifHCOutBroadcastPkts

The total number of packets that higher-level protocols requested be transmitted, and which 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.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.3.4.4.6 ifLinkUpDownTrapEnable

Indicates whether linkUp/linkDown traps should be generated for this interface.

- enabled(1)
- disabled(2)

By default, this object should have the value enabled(1) for interfaces which do not operate on “top” of any other interface (as defined in the ifStackTable), and disabled(2) otherwise.

Valid Object Value(s)

Table 31. ifLinkUpDownTrapEnable

Interface	Value
DSL Ports	**enabled(1)** or **disabled(2)**
DSL ATM Interfaces	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
DSL Ethernet Interfaces	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
Console Port	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
Ethernet Port 1 (10/100 Base T)	**enabled(1)** or **disabled(2)**
Ethernet Port 2 (10/100 Base T)	**enabled(1)** or **disabled(2)**
Ethernet Port 3 (1000 Base T / SFP)	**enabled(1)** or **disabled(2)**
Management In-Band Ethernet Interface	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
Management Out-of-Band Ethernet Interface	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
T1/E1 Ports	**enabled(1)** or **disabled(2)**
PPP HDLC Interface	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
PPP LCP Interface	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
ML PPP Bundle Interfaces	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**
PPP NCP Interface	The only valid value is **disabled(2)** Setting the object to any other value will result in **badValue**

5.3.4.4.7 ifHighSpeed

An estimate of the interface's current bandwidth in units of 1,000,000 bits per second. If this object reports a value of “n” then the speed of the interface is somewhere in the range of “n-500,000” to “n+499,999.” For interfaces which do not vary in bandwidth or for those where no accurate estimation can be made, this object should contain the nominal bandwidth. For a sub-layer which has no concept of bandwidth, this object should be zero.

Valid Object Value(s)

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM do not have any interfaces that exceed a Gauge32 syntax. As a result this object is not needed nor supported.

5.3.4.4.8 ifPromiscuousMode

This object has a value of false(2) if this interface only accepts packets/frames that are addressed to this station. This object has a value of true(1) when the station accepts all packets/frames transmitted on the media. The value true(1) is only legal on certain types of media. If legal, setting this object to a value of true(1) may require the interface to be reset before becoming effective.

The value of ifPromiscuousMode does not affect the reception of broadcast and multicast packets/frames by the interface.

Valid Object Value(s)

Table 32. ifPromiscuousMode

Interface	Value
DSL Ports	**true(1)**
DSL ATM Interfaces	**true(1)**
DSL Ethernet Interfaces	**true(1)**
Console Port	**true(1)**
Ethernet Port 1 (10/100 Base T)	**false(2)**
Ethernet Port 2 (10/100 Base T)	**false(2)**
Ethernet Port 3 (1000 Base T / SFP)	**false(2)**
Management In-Band Ethernet Interface	**true(1)**
Management Out-of-Band Ethernet Interface	**true(1)**
T1/E1 Ports	**true(1)**
PPP HDLC Interface	**true(1)**
PPP LCP Interface	**true(1)**
ML PPP Bundle Interfaces	**true(1)**
PPP NCP Interface	**true(1)**

5.3.4.4.9 ifConnectorPresent

- true(1) - if the interface sublayer has a physical connector
- false(2) - otherwise

Valid Object Value(s)

Table 33. ifConnectorPresent (1 of 2)

Interface	Value
DSL Ports	**true(1)**

Table 33. ifConnectorPresent (2 of 2)

Interface	Value
DSL ATM Interfaces	**false(2)**
DSL Ethernet Interfaces	**false(2)**
Console Port	**true(1)**
Ethernet Port 1 (10/100 Base T)	**true(1)**
Ethernet Port 2 (10/100 Base T)	**true(1)**
Ethernet Port 3 (1000 Base T / SFP)	**true(1)**
Management In-Band Ethernet Interface	**false(2)**
Management Out-of-Band Ethernet Interface	**false(2)**
T1/E1 Ports	**true(1)**
PPP HDLC Interface	**false(2)**
PPP LCP Interface	**false(2)**
ML PPP Bundle Interfaces	**false(2)**
PPP NCP Interface	**false(2)**

5.3.4.4.10 ifAlias

This object is an “alias” name for the interface as specified by a network manager, and provides a non-volatile 'handle' for the interface.

On the first instantiation of an interface, the value of ifAlias associated with that interface is the zero-length string. As and when a value is written into an instance of ifAlias through a network management set operation, then the agent must retain the supplied value in the ifAlias instance associated with the same interface for as long as that interface remains instantiated, including across all re- initializations/reboots of the network management system, including those which result in a change of the interface's ifIndex value.

An example of the value which 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 which 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.

5.3.4.4.11 ifCounterDiscontinuityTime

The value of sysUpTime on the most recent occasion at which any one or more of this interface's counters suffered a discontinuity. The relevant counters are the specific instances associated with this interface of any Counter32 or Counter64 object contained in the ifTable or ifXTable. If no such discontinuities have occurred since the last reinitialization of the local management subsystem, then this object contains a zero value.

5.3.4.5 ifStackTable

The table containing information on the relationships between the multiple sub-layers of network interfaces. In particular, it contains information on which sub-layers run 'on top of' which other sub-layers, where each sub-layer corresponds to a conceptual row in the ifTable. For example, when the sub-layer with ifIndex value x runs over the sub-layer with ifIndex value y, then this table contains:

- ifStackStatus.x.y=active

For each ifIndex value, I, which identifies an active interface, there are always at least two instantiated rows in this table associated with I. For one of these rows, I is the value of ifStackHigherLayer; for the other, I is the value of ifStackLowerLayer. (If I is not involved in multiplexing, then these are the only two rows associated with I.)

For example, two rows exist even for an interface which has no others stacked on top or below it:

- ifStackStatus.0.x=active
- ifStackStatus.x.0=active

This object is a sequence of ifStackEntry which contains information on a particular relationship between two sub-layers, specifying that one sub-layer runs on “top” of the other sub-layer. Each sub-layer corresponds to a conceptual row in the ifTable.

Indices:

- ifStackHigherLayer
- ifStackLowerLayer

OIDs:

- ifStackTable - { ifMIBObjects 2 } 1.3.6.1.2.1.31.1.2
- ifStackEntry - { ifStackTable 1 } 1.3.6.1.2.1.31.1.2.1

Table 34. ifStackTable

Object	OID	Syntax	Access	Status	Supported
ifStackHigherLayer	{ ifStackEntry 1 }	InterfaceIndexOrZero	not-accessible	current	Y
ifStackLowerLayer	{ ifStackEntry 2 }	InterfaceIndexOrZero	not-accessible	current	Y
ifStackStatus	{ ifStackEntry 3 }	RowStatus	read-create	current	Y

Valid Object Value(s)

Table 35. ifStackTable Values (1 of 2)

Interface[a]	Values[b]		Comments
	ifStackHigherLayer	ifStackLowerLayer	
DSL Ports	0	3 aaa[c]	Top of Stack, DSL Eth
	1 aaa[c]	0	Bottom of Stack, DSL
	2 aaa[c]	1 aaa[c]	DSL ATM over DSL
	3 aaa[c]	2 aaa[c]	DSL Eth over DSL ATM
Console Port	0	1 101	Only 1 layer
	1 101	0	
Ethernet Port 1 (10/100 Base T)	0	1 201	Only 1 layer
	1 201	0	
Ethernet Port 2 (10/100 Base T)	0	1 202	Only 1 layer
	1 202	0	

Table 35. ifStackTable Values (2 of 2)

Interface[a]	Values[b]		Comments
	ifStackHigherLayer	ifStackLowerLayer	
Ethernet Port 3 (1000 Base T / SFP)	0	1 203	Only 1 layer
	1 203	0	
Management In-Band Ethernet Interface	0	1 301	Only 1 layer
	1 301	0	
Management Out-of-Band Ethernet Interface	0	1 302	Only 1 layer
	1 302	0	
T1/E1 Ports (with ML PPP)	0	5 bbb[d]	Top of Stack, NCP
	5 bbb	4 bbb	NCP Bridge over ML PPP
	4 bbb	3 ccc[d]	ML PPP over LCP
	3 ccc	2 ccc	LCP over HDLC
	2 ccc	1 ccc	HDLC over T1/E1
	1 ccc	0	Bottom of Stack, T1/E1
T1/E1 Ports (without ML PPP)	0	5 bbb[d]	Top of Stack, NCP Bridge
	5 bbb	3 ccc[d]	NCP over LCP
	3 ccc	2 ccc	LCP over HDLC
	2 ccc	1 ccc	HDLC over T1/E1
	1 ccc	0	Bottom of Stack, T1/E1

a. Refer to Interface List, Table 11 on page 29, to determine when the interface or port is applicable.

b. Reference Section 5.3.4.3.1, Page 37, for a description of the ifIndex encoding scheme along with specific values of the ifIndex.
Spaces added and only the Layer and Port/Interface fields are displayed for clarification.

c. aaa refers to the associated DSL Port (001…048)

d. The port number bbb and ccc are independent.

5.3.4.5.1 ifStackHigherLayer

The value of ifIndex corresponding to the higher sub-layer of the relationship, that is, the sub-layer which runs on "top" of the sub-layer identified by the corresponding instance of ifStackLowerLayer. If there is no higher sub-layer (below the internetwork layer), then this object has the value 0.

5.3.4.5.2 ifStackLowerLayer

The value of ifIndex corresponding to the lower sub-layer of the relationship, that is, the sub-layer which runs "below" the sub-layer identified by the corresponding instance of ifStackHigherLayer. If there is no lower sub-layer, then this object has the value 0.

5.3.4.5.3 ifStackStatus

The status of the relationship between two sub-layers.

Changing the value of this object from “active” to “notInService” or “destroy” will likely have consequences up and down the interface stack. Thus, write access to this object is likely to be inappropriate for some types of interfaces, and many implementations will choose not to support write-access for any type of interface.

Valid Object Value(s)

Write-access is **not** supported.

The createAndWait RowStatus (Section 4.6, Page 13) is not supported.

5.3.4.5.4 ifStackLastChange

The value of sysUpTime at the time of the last change of the (whole) interface stack. A change of the interface stack is defined to be any creation, deletion, or change in value of any instance of ifStackStatus. If the interface stack has been unchanged since the last re-initialization of the local network management subsystem, then this object contains a zero value.

5.3.4.6 ifRcvAddressTable

The Generic Receive Address Table contains an entry for each address (broadcast, multicast, or unicast) for which the system will receive packets/frames on a particular interface, except as follows:

- for an interface operating in promiscuous mode, entries are only required for those addresses for which the system would receive frames were it not operating in promiscuous mode
- for 802.5 functional addresses, only one entry is required, for the address which has the functional address bit ANDed with the bit mask of all functional addresses for which the interface will accept frames

A system is normally able to use any unicast address which corresponds to an entry in this table as a source address.

This table replaces the ifExtnsRcvAddr table. The main difference is that this table makes use of the RowStatus (Section 4.6, Page 13) textual convention, while ifExtnsRcvAddr did not.

The table is a sequence of ifRcvAddressEntry which contains a list of objects identifying an address for which the system will accept packets/frames on the particular interface identified by the index value ifIndex.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- ifRcvAddressAddress

OIDs:

- ifRcvAddressTable - { ifMIBObjects 4 } 1.3.6.1.2.1.31.1.4
- ifRcvAddressEntry - { ifRcvAddressTable 1 } 1.3.6.1.2.1.31.1.4.1

Table 36. ifRcvAddressTable

Object	OID	Syntax	Access	Status	Supported
ifRcvAddressAddress	{ ifRcvAddressEntry 1 }	PhysAddress	not-accessible	current	Y
ifRcvAddressStatus	{ ifRcvAddressEntry 2 }	RowStatus	read-create	current	Y
ifRcvAddressType	{ ifRcvAddressEntry 3 }	INTEGER other(**1**) volatile(**2**) nonVolatile(**3**)	read-create	current	Y

5.3.4.6.1 ifRcvAddressAddress

An address for which the system will accept packets/frames on this entry's interface.

5.3.4.6.2 ifRcvAddressStatus

This object is used to create and delete rows in the ifRcvAddressTable.

Valid Object Value(s)

The createAndWait RowStatus (Section 4.6, Page 13) is not supported.

5.3.4.6.3 ifRcvAddressType

- other(1) - For those entries in the table which are valid and exist but are not classified as to whether they will continue to exist after the next restart
- volatile(2) - For those entries in the table which are valid and exist, but have not been saved, so that will not exist after the next restart of the managed system
- nonVolatile(3) - For those entries in the table which are valid and will not be deleted by the next restart of the managed system

The default value for this object is volatile(2).

5.3.4.7 ifTestTable

This table contains one entry per interface. It defines objects which allow a network manager to instruct an agent to test an interface for various faults. Tests for an interface are defined in the media-specific MIB for that interface. After invoking a test, the object ifTestResult can be read to determine the outcome. If an agent can not perform the test, ifTestResult is set to so indicate. The object ifTestCode can be used to provide further test- specific or interface-specific (or even enterprise-specific) information concerning the outcome of the test. Only one test can be in progress on each interface at any one time. If one test is in progress when another test is invoked, the second test is rejected. Some agents may reject a test when a prior test is active on another interface.

Before starting a test, a manager-station must first obtain “ownership” of the entry in the ifTestTable for the interface to be tested. This is accomplished with the ifTestId and ifTestStatus objects as follows:

```
   try_again:
   get (ifTestId, ifTestStatus)
   while (ifTestStatus != notInUse)
      /*
       * Loop while a test is running or some other
       * manager is configuring a test.
       */
      short delay
      get (ifTestId, ifTestStatus)
   }
   /*
    * Is not being used right now -- let's compete
    * to see who gets it.
    */
   lock_value = ifTestId
   if ( set(ifTestId = lock_value, ifTestStatus = inUse,
      ifTestOwner = 'my-IP-address') == FAILURE)
   /*
    * Another manager got the ifTestEntry -- go
    * try again
    */
   goto try_again;
/*
 * I have the lock
 */
set up any test parameters.
/*
 * This starts the test
 */
set(ifTestType = test_to_run);
wait for test completion by polling ifTestResult
when test completes, agent sets ifTestResult
     agent also sets ifTestStatus = "notInUse"
retrieve any additional test results, and ifTestId
if (ifTestId == lock_value+1) results are valid
```

A manager station first retrieves the value of the appropriate ifTestId and ifTestStatus objects, periodically repeating the retrieval if necessary, until the value of ifTestStatus is “notInUse.” The manager station then tries to set the same ifTestId object to the value it just retrieved, the same ifTestStatus object to “inUse,” and the corresponding ifTestOwner object to a value indicating itself. If the set operation succeeds then the manager has obtained ownership of the ifTestEntry, and the value of the ifTestId object is incremented by the agent (per the semantics of TestAndIncr). Failure of the set operation indicates that some other manager has obtained ownership of the ifTestEntry.

Once ownership is obtained, any test parameters can be setup, and then the test is initiated by setting ifTestType. On completion of the test, the agent sets ifTestStatus to “notInUse.” Once this occurs, the manager can retrieve the results. In the (rare) event that the invocation of tests by two network managers were to overlap, then there would be a possibility that the first test's results might be overwritten by the second test's results prior to the first results being read. This unlikely circumstance can be detected by a network manager retrieving ifTestId at the same time as retrieving the test results, and ensuring that the results are for the desired request.

If ifTestType is not set within an abnormally long period of time after ownership is obtained, the agent should time-out the manager, and reset the value of the ifTestStatus object back to “notInUse.” It is suggested that this time-out period be 5 minutes.

In general, a management station must not retransmit a request to invoke a test for which it does not receive a response; instead, it properly inspects an agent's MIB to determine if the invocation was successful. Only if the invocation was unsuccessful, is the invocation request retransmitted.

Some tests may require the interface to be taken off-line in order to execute them, or may even require the agent to reboot after completion of the test. In these circumstances, communication with the management station invoking the test may be lost until after completion of the test. An agent is not required to support such tests. However, if such tests are supported, then the agent should make every effort to transmit a response to the request which invoked the test prior to losing communication. When the agent is restored to normal service, the results of the test are properly made available in the appropriate objects. Note that this requires that the ifIndex value assigned to an interface must be unchanged even if the test causes a reboot. An agent must reject any test for which it cannot, perhaps due to resource constraints, make available at least the minimum amount of information after that test completes.

This table contains one entry per interface. It defines objects which allow a network manager to instruct an agent to test an interface for various faults.

This object is a sequence of ifTestEntry object which are objects for invoking tests on an interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- ifTestTable - { ifMIBObjects 3 } 1.3.6.1.2.1.31.1.3
- ifTestEntry - { ifTestTable 1 } 1.3.6.1.2.1.31.1.3.1

Table 37. ifTestTable

Object	OID	Syntax	Access	Status	Supported
ifTestId	{ ifTestEntry 1 }	TestAndIncr	read-only	deprecated	N
ifTestStatus	{ ifTestEntry 2 }	INTEGER notInUse(**1**) inUse(**2**)	read-write	deprecated	N
ifTestType	{ ifTestEntry 3 }	AutonomousType	read-write	deprecated	N
ifTestResult	{ ifTestEntry 4 }	INTEGER none(**1**) success(**2**) inProgress(**3**) notSupported(**4**) unAbleToRun(**5**) aborted(**6**) failed(**7**)	read-only	deprecated	N
ifTestCode	{ ifTestEntry 5 }	OBJECT IDENTIFIER	read-only	deprecated	N
ifTestOwner	{ ifTestEntry 6 }	OwnerString	read-write	deprecated	N

5.3.4.7.1 ifTestId

This object identifies the current invocation of the interface's test.

Valid Object Value(s)

This object is supported for a test trap and its access is restricted to read-only.

5.3.4.7.2 ifTestStatus

This object indicates whether or not some manager currently has the necessary “ownership” required to invoke a test on this interface. A write to this object is only successful when it changes its value from “notInUse(1)” to “inUse(2).” After completion of a test, the agent resets the value back to “notInUse(1).

5.3.4.7.3 ifTestType

A control variable used to start and stop operator- initiated interface tests. Most OBJECT IDENTIFIER values assigned to tests are defined elsewhere, in association with specific types of interface. However, this document assigns a value for a full-duplex loopback test, and defines the special meanings of the subject identifier:

- noTest OBJECT IDENTIFIER = { 0 0 }

When the value noTest is written to this object, no action is taken unless a test is in progress, in which case the test is aborted. Writing any other value to this object is only valid when no test is currently in progress, in which case the indicated test is initiated.

When read, this object always returns the most recent value that ifTestType was set to. If it has not been set since the last initialization of the network management subsystem on the agent, a value of noTest is returned.

5.3.4.7.4 ifTestResult

This object contains the result of the most recently requested test, or the value none(1) if no tests have been requested since the last reset. Note that this facility provides no provision for saving the results of one test when starting another, as could be required if used by multiple managers concurrently. Values for this object are:

- none(1) - no test yet requested
- success(2)
- inProgress(3)
- notSupported(4)
- unAbleToRun(5) - due to state of system
- aborted(6)
- failed(7)

5.3.4.7.5 ifTestCode

This object contains a code which contains more specific information on the test result, for example an error-code after a failed test. Error codes and other values this object may take are specific to the type of interface and/or test. The value may have the semantics of either the AutonomousType or InstancePointer textual conventions as defined in RFC 2579. The identifier:

- testCodeUnknown OBJECT IDENTIFIER = { 0 0 }

is defined for use if no additional result code is available.

5.3.4.7.6 ifTestOwner

The entity which currently has the “ownership” required to invoke a test on this interface.

5.3.5 Traps

OIDs

- snmpTraps - { snmpMIBObjects 5 } 1.3.6.1.6.3.1.1.5

Table 38. Interface Group Traps

Object	OID	Trap Objects	Status	Supported
linkDown	{ snmpTraps 3 }	ifIndex ifAdminStatus ifOperStatus	current	Y
linkUp	{ snmpTraps 4 }	ifIndex ifAdminStatus ifOperStatus	current	Y

5.3.5.1 linkDown

A linkDown trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links is about to enter the down state from some other state (but not from the notPresent state). This other state is indicated by the included value of ifOperStatus.

Objects:

- ifIndex
- ifAdminStatus
- ifOperStatus

5.3.5.2 linkUp

A linkUp trap signifies that the SNMP entity, acting in an agent role, has detected that the ifOperStatus object for one of its communication links left the down state and transitional into some other state (but not into the notPresent state). This other state is indicated by the included value of ifOperStatus.

Objects:

- ifIndex
- ifAdminStatus
- ifOperStatus

5.4 IP-MIB — RFC 2011

5.4.1 Introduction

This MIB describes objects for managing implementations of the Internet Protocol (IP) and its associated Internet Control Message Protocol (ICMP). This MIB excludes the management of IP routes.

The RFC used as a reference was dated November 1996. The LAST-UPDATED for the IP-MIB was “9411010000z.”

5.4.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - { mib-2 4 } 1.3.6.1.2.1.4
- icmp - { mib-2 5 } 1.3.6.1.2.1.5
- ipMIB - { mib-2 48 } 1.3.6.1.2.1.48
- ipRouteTable - { ip 21 } 1.3.6.1.2.1.4.21

5.4.3 MIB Objects

The following groups supported in this MIB are as follow:

- IP Group
- ICMP Group

5.4.3.1 IP Group

OIDs:

- ip - { mib-2 4} 1.3.6.1.2.1.4

Table 39. IP-MIB Objects (1 of 2)

Object	OID	Syntax	Access	Status	Supported
ipForwarding	{ ip 1 }	INTEGER forwarding(**1**) not-forwarding(**2**)	read-write	current	Y
ipDefaulTTL	{ ip 2 }	INTEGER(1…255)	read-write	current	Y
ipInReceives	{ ip 3 }	Counter32	read-only	current	Y
ipInHdrErrors	{ ip 4 }	Counter32	read-only	current	N
ipInAddrErrors	{ ip 5 }	Counter32	read-only	current	N
ipForwDatagrams	{ ip 6 }	Counter32	read-only	current	N
ipInUnknownProtos	{ ip 7 }	Counter32	read-only	current	N
ipInDiscards	{ ip 8 }	Counter32	read-only	current	N

Table 39. IP-MIB Objects (2 of 2)

Object	OID	Syntax	Access	Status	Supported
ipInDelivers	{ ip 9 }	Counter32	read-only	current	N
ipOutRequests	{ ip 10 }	Counter32	read-only	current	N
ipOutDiscards	{ ip 11 }	Counter32	read-only	current	N
ipOutNoRoutes	{ ip 12 }	Counter32	read-only	current	N
ipReasmTimeout	{ ip 13 }	Integer32	read-only	current	N
ipReasmReqds	{ ip 14 }	Counter32	read-only	current	N
ipReasmOKs	{ ip 15 }	Counter32	read-only	current	N
ipReasmFails	{ ip 16 }	Counter32	read-only	current	N
ipFragOKs	{ ip 17 }	Counter32	read-only	current	N
ipFragFails	{ ip 18 }	Counter32	read-only	current	N
ipFragCreates	{ ip 19 }	Counter32	read-only	current	N
ipAddrTable	{ ip 20 }	Sequence of ipAddrEntry	not-accessible	current	Y

5.4.3.1.1 ipForwarding

The indication of whether this entity is acting as an IP router in respect to the forwarding of datagrams received by, but not addressed to, this entity. IP routers forward datagrams. IP hosts do not (except those source-routed via the host). Values for this object are:

- forwarding(1) - acting as a router
- notForwarding(2) - NOT acting as a router

Valid Object Value(s)

The only value supported for this object is notForwarding(2).

5.4.3.1.2 ipDefaultTTL

The default value inserted into the Time-To-Live field of the IP header of datagrams originated at this entity, whenever a TTL value is not supplied by the transport layer protocol.

The default value for this object is 255.

5.4.3.1.3 ipInReceives

The total number of input datagrams received from interfaces, including those received in error.

5.4.3.1.4 ipInHdrErrors

The number of input datagrams discarded due to errors in their IP headers, including bad checksums, version number mismatch, other format errors, time-to-live exceeded, errors discovered in processing their IP options, etc.

5.4.3.1.5 ipInAddrErrors

The number of input datagrams discarded because the IP address in their IP header's destination field was not a valid address to be received at this entity. This count includes invalid addresses (for example, 0.0.0.0) and addresses of

unsupported Classes (for example, Class E). For entities which are not IP routers and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address.

5.4.3.1.6 ipForwDatagrams

The number of input datagrams for which this entity was not their final IP destination, as a result of which an attempt was made to find a route to forward them to that final destination. In entities which do not act as IP routers, this counter will include only those packets which were Source-Routed via this entity, and the Source-Route option processing was successful.

5.4.3.1.7 ipInUnknownProtos

The number of locally-addressed datagrams received successfully but discarded because of an unknown or unsupported protocol.

5.4.3.1.8 ipInDiscards

The number of input IP datagrams for which no problems were encountered to prevent their continued processing, but which were discarded (for example, for lack of buffer space). Note that this counter does not include any datagrams discarded while awaiting re-assembly.

5.4.3.1.9 ipInDelivers

The total number of input datagrams successfully delivered to IP user-protocols (including ICMP).

5.4.3.1.10 ipOutRequests

The total number of IP datagrams which local IP user-protocols (including ICMP) supplied to IP in requests for transmission. Note that this counter does not include any datagrams counted in ipForwDatagrams.

5.4.3.1.11 ipOutDiscards

The number of output IP datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (for example, for lack of buffer space). Note that this counter would include datagrams counted in ipForwDatagrams if any such packets met this (discretionary) discard criterion.

5.4.3.1.12 ipOutNoRoutes

The number of IP datagrams discarded because no route could be found to transmit them to their destination. Note that this counter includes any packets counted in ipForwDatagrams which meet this `no-route' criterion. Note that this includes any datagrams which a host cannot route because all of its default routers are down.

5.4.3.1.13 ipReasmTimeout

The maximum number of seconds which received fragments are held while they are awaiting reassembly at this entity.

This object is set to 60.

5.4.3.1.14 ipReasmReqds

The number of IP fragments received which needed to be reassembled at this entity.

5.4.3.1.15 ipReasmOKs

The number of IP datagrams successfully re-assembled.

5.4.3.1.16 ipReasmFails

The number of failures detected by the IP re-assembly algorithm (for whatever reason: timed out, errors, etc.). Note that this is not necessarily a count of discarded IP fragments since some algorithms (notably the algorithm in RFC 815) can lose track of the number of fragments by combining them as they are received.

5.4.3.1.17 ipFragOKs

The number of IP datagrams that have been successfully fragmented at this entity.

5.4.3.1.18 ipFragFails

The number of IP datagrams that have been discarded because they needed to be fragmented at this entity but could not be, for example, because their Don't Fragment flag was set.

5.4.3.1.19 ipFragCreates

The number of IP datagram fragments that have been generated as a result of fragmentation at this entity.

5.4.3.1.20 ipAddrTable

The table of addressing information relevant to this entity's IP addresses. This table is a sequence of ipAddrEntry which contains the addressing information for one of this entity's IP addresses.

Index:

- ipAdEntAddr

OIDs:

- ipAddrTable - { ip 20 } 1.3.6.1.2.1.4.20
- ipAddrEntry - {ipAddrTable 1 } 1.3.6.1.2.1.4.20.1

Table 40. ipAddrTable

Object	OID	Syntax	Access	Status	Supported
ipAdEntAddr	{ ipAddrEntry 1 }	IpAddress	read-only	current	Y
ipAdEntIfIndex	{ ipAddrEntry 2 }	INTEGER (1…2147483674)	read-only	current	Y
ipAdEntNetMask	{ ipAddrEntry 3 }	IpAddress	read-only	current	Y
ipAdEntBcastAddr	{ ipAddrEntry 4 }	INTEGER(0…1)	read-only	current	Y
ipAdEntReasmMaxSize	{ ipAddrEntry 5 }	INTEGER(0…65535)	read-only	current	Y

5.4.3.1.20.1 ipAdEntAddr

The IP address to which this entry's addressing information pertains.

5.4.3.1.20.2 ipAdEntIfIndex

The index value which uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface as identified by the same value of RFC 1573's ifIndex.

5.4.3.1.20.3 ipAdEntNetMask

The subnet mask associated with the IP address of this entry. The value of the mask is an IP address with all the network bits set to 1 and all the hosts bits set to 0.

5.4.3.1.20.4 ipAdEntBcastAddr

The value of the least-significant bit in the IP broadcast address used for sending datagrams on the (logical) interface associated with the IP address of this entry. For example, when the Internet standard all-ones broadcast address is used, the value is 1. This value applies to both the subnet and network broadcasts addresses used by the entity on this (logical) interface.

5.4.3.1.20.5 ipAdEntReasmMaxSize

The size of the largest IP datagram which this entity can re-assemble from incoming IP fragmented datagrams received on this interface.

Valid Object Value(s)

1500

5.4.3.1.21 ipRouteTable

The IP Routing Table is deprecated by the Classless Inter-Domain Routing (CIDR) Table, ipCidrRouteTable, specified in RFC 2096.

5.4.3.1.22 ipNetToMediaTable

The IP Address Translation tables contain the IpAddress to “physical” address equivalences. Some interfaces do not use translation tables for determining address equivalences (for example, DDN-X.25 has an algorithmic method); if all interfaces are of this type, then the Address Translation table is empty, that is, has zero entries.

The IP Address Translation table used for mapping from IP addresses to physical addresses.

The ipNetToMediaTable is a sequence of ipNetToMedialEntry which contains one ipAddress to physical address equivalence.

Indices:

- ipNetToMediaIfIndex
- ipNetToMediaNetAddress

OIDs:

- ipNetToMediaTable - { ip 22 } 1.3.6.1.2.1.4.22
- ipNetToMediaEntry - { ipNetToMediaTable 1 } 1.3.6.1.2.1.4.22.1

Table 41. ipNetToMediaTable

Object	OID	Syntax	Access	Status	Supported
ipNetToMediaIfIndex	{ ipNetToMediaEntry 1 }	INTEGER (1…2147483674)	read-write	current	Y
ipNetToMediaPhysAddress	{ ipNetToMediaEntry 2 }	PhysAddress	read-write	current	Y
ipNetToMediaNetAddress	{ ipNetToMediaEntry 3 }	IpAddress	read-write	current	Y
ipNetToMediaType	{ ipNetToMediaEntry 4 }	INTEGER other(1) invalid(2) dynamic(3) static(4)	read-write	current	Y

5.4.3.1.22.1 ipNetToMediaIfIndex

The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of RFC 1573's ifIndex.

5.4.3.1.22.2 ipNetToMediaPhysAddress

The media-dependent “physical” address.

5.4.3.1.22.3 ipNetToMediaNetAddress

The IpAddress corresponding to the media-dependent “physical” address.

5.4.3.1.22.4 ipNetToMediaType

The values for this object are:

- other(1) - None of the following
- invalid(2) - An invalidated mapping
- dynamic(3)
- static(4)

The type of mapping. Setting this object to the value invalid(2) has the effect of invalidating the corresponding entry in the ipNetToMediaTable. That is, it effectively disassociates the interface identified with said entry from the mapping identified with said entry. It is an implementation- specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use. Proper interpretation of such entries requires examination of the relevant ipNetToMediaType object.

5.4.3.1.23 ipRoutingDiscards

The number of routing entries which were chosen to be discarded even though they are valid. One possible reason for discarding such an entry could be to free-up buffer space for other routing entries.

5.4.3.2 ICMP Group

OIDs:

- icmp - { mib-2 5 } 1.3.6.1.2.1.5

Table 42. ICMP Group (1 of 2)

Object	OID	Syntax	Access	Status	Supported
icmpInMsgs	{ icmp 1 }	Counter32	read-only	current	N
icmpInErrors	{ icmp 2 }	Counter32	read-only	current	N
icmpInDestUnreachs	{ icmp 3 }	Counter32	read-only	current	N
icmpInTimeExcds	{ icmp 4 }	Counter32	read-only	current	N
icmpInParmProbs	{ icmp 5 }	Counter32	read-only	current	N
icmpInSrcQuenchs	{ icmp 6 }	Counter32	read-only	current	N
icmpInRedirects	{ icmp 7 }	Counter32	read-only	current	N
icmpInEchos	{ icmp 8 }	Counter32	read-only	current	N
icmpInEchoReps	{ icmp 9 }	Counter32	read-only	current	N
icmpInTimestamps	{ icmp 10 }	Counter32	read-only	current	N
icmpInTimestampReps	{ icmp 11 }	Counter32	read-only	current	N
icmpInAddrMasks	{ icmp 12 }	Counter32	read-only	current	N
icmpInAddrMaskReps	{ icmp 13 }	Counter32	read-only	current	N
icmpOutMsgs	{ icmp 14 }	Counter32	read-only	current	N
icmpOutErrors	{ icmp 15 }	Counter32	read-only	current	N
icmpOutDestUnreachs	{ icmp 16 }	Counter32	read-only	current	N

Table 42. ICMP Group (2 of 2)

Object	OID	Syntax	Access	Status	Supported
icmpOutTimeExcds	{ icmp 17 }	Counter32	read-only	current	N
icmpOutParmProbs	{ icmp 18 }	Counter32	read-only	current	N
icmpOutSrcQuenchs	{ icmp 19 }	Counter32	read-only	current	N
icmpOutRedirects	{ icmp 20 }	Counter32	read-only	current	N
icmpOutEchos	{ icmp 21 }	Counter32	read-only	current	N
icmpOutEchoReps	{ icmp 22 }	Counter32	read-only	current	N
icmpOutTimestamps	{ icmp 23 }	Counter32	read-only	current	N
icmpOutTimestampReps	{ icmp 24 }	Counter32	read-only	current	N
icmpOutAddrMasks	{ icmp 25 }	Counter32	read-only	current	N
icmpOutAddrMaskReps	{ icmp 26 }	Counter32	read-only	current	N

5.4.3.2.1 icmpInMsgs

The total number of ICMP messages which the entity received. Note that this counter includes all those counted by icmpInErrors.

5.4.3.2.2 icmpInErrors

The number of ICMP messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.).

5.4.3.2.3 icmpInDestUnreachs

The number of ICMP Destination Unreachable messages received.

5.4.3.2.4 icmpInTimeExcds

The number of ICMP Time Exceeded messages received.

5.4.3.2.5 icmpInParmProbs

The number of ICMP Parameter Problem messages received.

5.4.3.2.6 icmpInSrcQuenchs

The number of ICMP Source Quench messages received.

5.4.3.2.7 icmpInRedirects

The number of ICMP Redirect messages received.

5.4.3.2.8 icmpInEchos

The number of ICMP Echo (request) messages received.

5.4.3.2.9 icmpInEchoReps

The number of ICMP Echo Reply messages received.

5.4.3.2.10 icmpInTimestamps

The number of ICMP Timestamp (request) messages received.

5.4.3.2.11 icmpInTimestampReps

The number of ICMP Timestamp Reply messages received.

5.4.3.2.12 icmpInAddrMasks

The number of ICMP Address Mask Request messages received.

5.4.3.2.13 icmpInAddrMaskReps

The number of ICMP Address Mask Reply messages received.

5.4.3.2.14 icmpOutMsgs

The total number of ICMP messages which this entity attempted to send. Note that this counter includes all those counted by icmpOutErrors.

5.4.3.2.15 icmpOutErrors

The number of ICMP messages which this entity did not send due to problems discovered within ICMP such as a lack of buffers. This value should not include errors discovered outside the ICMP layer such as the inability of IP to route the resultant datagram. In some implementations there may be no types of error which contribute to this counter's value.

5.4.3.2.16 icmpOutDestUnreachs

The number of ICMP Destination Unreachable messages sent.

5.4.3.2.17 icmpOutTimeExcds

The number of ICMP Time Exceeded messages sent.

5.4.3.2.18 icmpOutParmProbs

The number of ICMP Parameter Problem messages sent.

5.4.3.2.19 icmpOutSrcQuenchs

The number of ICMP Source Quench messages sent.

5.4.3.2.20 icmpOutRedirects

The number of ICMP Redirect messages sent. For a host, this object will always be zero, since hosts do not send redirects.

5.4.3.2.21 icmpOutEchos

The number of ICMP Echo (request) messages sent.

5.4.3.2.22 icmpOutEchoReps

The number of ICMP Echo Reply messages sent.

5.4.3.2.23 icmpOutTimestamps

The number of ICMP Timestamp (request) messages sent.

5.4.3.2.24 icmpOutTimestampReps

The number of ICMP Timestamp Reply messages sent.

5.4.3.2.25 icmpOutAddrMasks

The number of ICMP Address Mask Request messages sent.

5.4.3.2.26 icmpOutAddrMaskReps

The number of ICMP Address Mask Reply messages sent.

5.5 IP-FORWARD-MIB — RFC 2096

5.5.1 Introduction

This MIB is for displaying of Classless Inter-Domain Routing (CIDR) multipath IP Routes. The RFC used as a reference was dated January 1997. The LAST-UPDATED for the IP_FORWARD-MIB was “9609190000Z.”

5.5.2 OIDs

- iso - { 1 }
- org - { iso 3 }
- dod - { org 6 }
- internet - { dod 1 }
- mgmt - { internet 2 }
- mib-2 - { mgmt 1 }
- ip - { mib-2 4 }
- ipForward - { ip 24 }
- ipForwardNumber - { ipForward 1 } - Obsoleted by ipCidrRouteNumber
- ipForwardTable - { ipForward 2 } - Obsoleted by ipCidrRouteTable
- ipCidrRouteNumber - { ipForward 3 }
- ipCidrRouteTable - { ipForward 4 }

5.5.3 MIB Objects

Table 43. IP-FORWARD-MIB Objects

Object	OID	Syntax	Access	Status	Supported
ipCidrRouteNumber	{ ipForward 3 }	Gauge32	read-only	current	Y
ipCidrRouteTable	{ ipForward 4 }	Sequence of ipCidrRouteEntry	not-accessible	current	Y

5.5.3.1 ipCidrRouteNumber

The number of current ipCidrRouteTable entries.

OID:

- ipCidrRouteNumber - { ipForward 3 }

5.5.3.2 ipCidrRouteTable

The IP CIDR Route Table obsoletes and replaces the ipRoute Table current in MIB-I and MIB-II and the IP Forwarding Table (RFC 1354). It adds knowledge of the autonomous system of the next hop, multiple next hop support, and policy routing, and Classless Inter-Domain Routing.

Indices:

- ipCidrRouteDest
- ipCidrRouteMask
- ipCidrRouteTos
- ipCidrRouteNextHop

OIDs:

- ipCidrRouteTable - { ipForward 4 }
- ipCidrRouteEntry - { ipCidrRouteTable 1 }

Table 44. ipCidrRouteTable

Object	OID	Syntax	Access	Status	Supported
ipCidrRouteDest	{ ipCidrRouteEntry 1 }	IpAddress	read-only	current	Y
ipCidrRouteMask	{ ipCidrRouteEntry 2 }	IpAddress	read-only	current	Y
ipCidrRouteTos	{ ipCidrRouteEntry 3 }	Integer32	read-only	current	Y
ipCidrRouteNextHop	{ ipCidrRouteEntry 4 }	IpAddress	read-only	current	Y
ipCidrRouteIfIndex	{ ipCidrRouteEntry 5 }	Integer32	read-create	current	Y
ipCidrRouteType	{ ipCidrRouteEntry 6 }	INTEGER other(1) reject(2) local(3) remote(4)	read-create	current	N
ipCidrRouteProto	{ ipCidrRouteEntry 7 }	INTEGER other(1) local(2) netmgmt(3) icmp(3) egp(5) ggp(6) hello(7) rip(8) isIs(9) esIs(10) ciscoIgrp(11) bbnSpfIgp(12) ospf(13) bgp(14) idpr(15) ciscoEigrp(16)	read-only	current	Y
ipCidrRouteAge	{ ipCidrRouteEntry 8 }	Integer32	read-only	current	N
ipCidrRouteInfo	{ ipCidrRouteEntry 9 }	OBJECT IDENTIFIER	read-create	current	N
ipCidrRouteNextHopAS	{ ipCidrRouteEntry 10 }	Integer32	read-create	current	N
ipCidrRouteMetric1	{ ipCidrRouteEntry 11 }	Integer32	read-create	current	N
ipCidrRouteMetric2	{ ipCidrRouteEntry 12 }	Integer32	read-create	current	N
ipCidrRouteMetric3	{ ipCidrRouteEntry 13 }	Integer32	read-create	current	N
ipCidrRouteMetric4	{ ipCidrRouteEntry 14 }	Integer32	read-create	current	N
ipCidrRouteMetric5	{ ipCidrRouteEntry 15 }	Integer32	read-create	current	N
ipCidrRouteStatus	{ ipCidrRouteEntry 16 }	RowStatus	read-create	current	Y

5.5.3.2.1 ipCidrRouteDest

The destination IP address of this route.

This object may not take a Multicast (Class D) address value.

Any assignment (implicit or otherwise) of an instance of this object to a value x must be rejected if the bitwise logical-AND of x with the value of the corresponding instance of the ipCidrRouteMask object is not equal to x.

5.5.3.2.2 ipCidrRouteMask

Indicate the mask to be logical-ANDed with the destination address before being compared to the value in the ipCidrRouteDest field. For those systems that do not support arbitrary subnet masks, an agent constructs the value of the ipCidrRouteMask by reference to the IP Address Class.

Any assignment (implicit or otherwise) of an instance of this object to a value x must be rejected if the bitwise logical-AND of x with the value of the corresponding instance of the ipCidrRouteDest object is not equal to ipCidrRouteDest.

5.5.3.2.3 ipCidrRouteTos

The policy specifier is the IP TOS Field. The encoding of IP TOS is as specified by the following convention. Zero indicates the default path if no more specific policy applies.

The following convention is included for specification of TOS Field contents. At this time, the Host Requirements and the Router Requirements documents disagree on the width of the TOS field. This mapping describes the Router Requirements mapping, and leaves room to widen the TOS field without impact to fielded systems.

Table 45. ipCidrRouteTos

Precedence			Type of Service				0	Policy Code
			0	0	0	0	0	0
			0	0	0	1	0	2
			0	0	1	0	0	4
			0	0	1	1	0	6
			0	1	0	0	0	8
			0	1	0	1	0	10
			0	1	1	0	0	12
			0	1	1	1	0	14
			1	0	0	0	0	16
			1	0	0	1	0	18
			1	0	1	0	0	20
			1	0	1	1	0	22
			1	1		0	0	24
			1	1	0	1	0	26
			1	1	1	0	0	28
			0	1	1	1	0	30

Valid Object Value(s)

This object is not used in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. However, it is supported because it is one of the indicies for the table. Therefore, only “0” is supported.

5.5.3.2.4 ipCidrRouteNextHop

On remote routes, the address of the next system en route; Otherwise, 0.0.0.0.

5.5.3.2.5 ipCidrRouteIfIndex

The ifIndex value which identifies the local interface through which the next hop of this route should be reached.

5.5.3.2.6 ipCidrRouteType

The type of route. Values for this object are listed below. Note that local(3) refers to a route for which the next hop is the final destination; remote(4) refers to a route for which the next hop is not the final destination.

Routes which do not result in traffic forwarding or rejection should not be displayed even if the implementation keeps them stored internally.

reject (2) refers to a route which, if matched, discards the message as unreachable. This is used in some protocols as a means of correctly aggregating routes.

The values for this object are:

- other(1) - not specified by this MIB
- reject(2) - route which discards traffic
- local(3) - local interface
- remote(4) - remote destination

5.5.3.2.7 ipCidrRouteProto

The routing mechanism via which this route was learned. Inclusion of values for gateway routing protocols is not intended to imply that hosts should support those protocols.

The values for this object are:

- other(1) - not specified
- local(2) - local interface
- netmgmt(3) - static route
- icmp(4) - result of ICMP Redirect

Note: The following are all dynamic routing protocols

- egp(5) - Exterior Gateway Protocol
- ggp(6) - Gateway-Gateway Protocol
- hello(7) - FuzzBall HelloSpeak
- rip (8) - Berkeley RIP or RIP-II
- isIs(9) - Dual IS-IS
- esIs(10) - ISO 9542
- ciscoIgrp(11) - Cisco IGRP
- bbnSpfIgp(12) - BBN SPF IGP
- ospf(13) - Open Shortest Path First
- bgp(14) - Border Gateway Protocol
- idpr(15) - InterDomain Policy Routing
- ciscoEigrp(16) - Cisco EIGRP

Valid Object Value(s)

Since the BitStorm 2600 and GranDSLAM 4200 IP DSLAM use this table to defined static routes, the only value supported is netMgmt(3).

5.5.3.2.8 ipCidrRouteAge

The number of seconds since this route was last updated or otherwise determined to be correct. Note that no semantics of ‘too old’ can be implied except through knowledge of the routing protocol by which the route was learned.

5.5.3.2.9 ipCidrRouteInfo

A reference to MIB definitions specific to the particular routing protocol which is responsible for this route, as date mined by the value specified in the route’s ipCidrRouteProto value. 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 implementation conforming to ASN.1 and the Basic Encoding Rules must be able to generate and recognize this value.

5.5.3.2.10 ipCidrRouteNextHopAS

The Autonomous System Number of the Next Hop. The semantics of this object are determined by the routing protocol specified in the route’s ipCidrRouteProto value. When this object is unknown or not relevant its value should be set to zero.

5.5.3.2.11 ipCidrRouteMetric1

The primary routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route’s ipCidrRouteProto value. If this metric is not used, its value should be set to -1.

5.5.3.2.12 ipCidrRouteMetric2

An alternate routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route’s ipCidrRouteProto value. If this metric is not used, its value should be set to -1.

5.5.3.2.13 ipCidrRouteMetric3

An alternate routing metric for this route. The semantics of this metric are determined by the routing protocol specified in the route’s ipCidrRouteProto value. If this metric is not used, its value should be set to -1.

5.5.3.2.14 ipCidrRouteMetric4

An alternate routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route’s ipCidrRouteProto value. If this metric is not used, its value should be set to -1.

5.5.3.2.15 ipCidrRouteMetric5

An alternate routing metric for this route. The semantics of this metric are determined by the routing-protocol specified in the route’s ipCidrRouteProto value. If this metric is not used, its value should be set to -1.

5.5.3.2.16 ipCidrRouteStatus

The row status variable, used according to row installation and removal conventions.

5.6 ENTITY-MIB (Version 2) — RFC 2737

5.6.1 Introduction

This MIB describes managed objects used for managing multiple logical and physical entities managed by a single SNMP agent. The RFC used as a reference was dated December 1999. The LAST-UPDATED for this MIB in the RFC was “9912070000Z.”

5.6.2 Terms

The following are terms used throughout this MIB:

- Naming Scope - A “naming scope” represents the set of information that may be potentially accessed through a single SNMP operation. All instances within the naming scope share the same unique identifier space. For SNMPv1, a naming scope is identified by the value of the associated “entLogicalCommunity” instance. For SNMPv3, the term “context” is used instead of “naming scope.” The complete definition of an SNMP context can be found in section 3.3.1 of RFC 2571.

  Each BitStorm 2600 and GranDSLAM 4200 IP DSLAM have a single naming scope.
- Multi-Scoped Object - A MIB object, for which identical instance values identify different managed information in different naming scopes, is called a “multi-scoped” MIB object.
- Single-Scoped Object - A MIB object, for which identical instance values identify the same managed information in different naming scopes, is called a “single-scoped” MIB object.

  Each managed object in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM is single-scoped.
- Logical Entity - A managed system contains one or more logical entities, each represented by at most one instantiation of each of a particular set of MIB objects. A set of management functions is associated with each logical entity. Examples of logical entities include routers, bridges, print-servers, etc.
- Physical Entity - A “physical entity” or “physical component” represents an identifiable physical resource within a managed system. Zero or more logical entities may utilize a physical resource at any given time. It is an implementation-specific manner as to which physical components are represented by an agent in the EntPhysicalTable. Typically, physical resources (such as communications ports, backplanes, sensors, daughter-cards, power supplies, the overall chassis) which can be managed via functions associated with one or more logical entities are included in the MIB.
- Containment Tree - Each physical component may optionally be modeled as “contained” within another physical component. A “containment-tree” is the conceptual sequence of entPhysicalIndex values which uniquely specifies the exact physical location of a physical component within the managed system. It is generated by “following and recording” each “entPhysicalContainedIn” instance “up the tree towards the root,” until a value of zero indicating no further containment is found.

5.6.3 Textual Conventions

The following textual conventions are made in this RFC:

- PhysicalIndex - An arbitrary value which uniquely identifies the physical entity. The value is a small positive integer; index values for different physical entities are not necessarily contiguous.
  — SYNTAX INTEGER(1…2147483647).
- Physical Class - An enumerated value which provides an indication of the general hardware type of a particular physical entity.There are no restrictions as to the num-

ber of entPhysicalEntries of each entPhysicalClass, which must be instantiated by an agent.

— SYNTAX INTEGER

- other(1) - applicable if the physical entity class is known, but does not match any of the supported values.
- unknown(2) - applicable if the physical entity class is unknown to the agent.
- chassis(3) - applicable if the physical entity class is an overall container for networking equipment. Any class of physical entity except a stack may be contained within a chassis, and a chassis may only be contained within a stack.
- backplane(4) - applicable if the physical entity class is some sort of device for aggregating and forwarding networking traffic, such as a shared backplane in a modular Ethernet switch. Note that an agent may model a backplane as a single physical entity, which is actually implemented as multiple discrete physical components (within a chassis or stack).
- container(5) - applicable if the physical entity class is capable of containing one or more removable physical entities, possibly of different types. For example, each (empty or full) slot in a chassis is modeled as a container. Note that all removable physical entities should be modeled within a container entity, such as field-replaceable modules, fans, or power supplies. Note that all known containers should be modeled by the agent, including empty containers.
- powerSupply(6) - applicable if the physical entity class is a power-supplying component.
- fan(7) - applicable if the physical entity class is a fan or other heat-reduction component.
- sensor(8) - applicable if the physical entity class is some sort of sensor, such as a temperature sensor within a router chassis.
- module(9) - applicable if the physical entity class is some sort of self-contained sub-system. If it is removable, then it should be modeled within a container entity, otherwise it should be modeled directly within another physical entity (for example, a chassis or another module).
- port(10) - applicable if the physical entity class is some sort of networking port, capable of receiving and/or transmitting networking traffic.
- stack(11) - applicable if the physical entity class is some sort of super-container (possibly virtual), intended to group together multiple chassis entities. A stack may be realized by a 'virtual' cable, a real interconnect cable, attached to multiple chassis, or may in fact be comprised of multiple interconnect cables. A stack should not be modeled within any other physical entities,

but a stack may be contained within another stack. Only chassis entities should be contained within a stack.

5.6.4 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- entityMIB - { mib-2 47 } 1.3.6.1.2.1.47
- entityMIBObjects - { entityMIB 1 } 1.3.6.1.2.1.47.1
- entityPhysical - { entityMIBObjects 1 } 1.3.6.1.2.1.47.1.1
- entityLogical - { entityMIBObjects 2 } 1.3.6.1.2.1.47.1.2
- entityMapping - { entityMIBObjects 3 } 1.3.6.1.2.1.47.1.3
- entityGeneral - { entityMIBObjects 4 } 1.3.6.1.2.1.47.1.4
- entityMIBTraps - { entityMIB 2 } 1.3.6.1.2.1.47.1
- entityMIBTrapPrefix - { entityMIBTraps 0 } 1.3.6.1.2.1.47.1.0

5.6.5 MIB Objects

The following groups are defined in this MIB:

- entityPhysical group - Describes the physical entities managed by a single agent.
- entityLogical group - Describes the logical entities managed by a single agent. This group is not implemented.
- entityMapping group - Describes the associations between the physical entities, logical entities, interfaces, and non-interface ports managed by a single agent.
- entityGeneral group - Describes general system attributes shared by potentially all types of entities managed by a single agent.

Table 46. ENTITY-MIB Objects

Object	OID	Syntax	Access	Status	Supported
entPhysicalTable	{ entityPhysical 1 }	Sequence of entPhysicalEntry	not-accessible	current	Y
entLogicalTable	{ entityLogical 1 }	Sequence of entLogicalEntry	not-accessible	current	N
entLPMappingTable	{ entityMapping 1 }	Sequence of entLPMappingEntry	not-accessible	current	N
entAliasMappingTable	{ entityMapping 2 }	Sequence of entAliasMappingEntry	not-accessible	current	Y
entPhysicalContainsTable	{ entityMapping 3 }	Sequence of entPhysicalContainsEntry	not-accessible	current	Y
entLastChangeTime	{ entityGeneral 1 }	TimeStamp	read-only	current	Y

5.6.5.1 entityPhysical Group

This group contains a single table to identify physical system components, called the entPhysicalTable.

The entPhysicalTable contains one row per physical entity, and must always contain at least one row for an “overall” physical entity, which should have an entPhysicalClass value of “stack(11),” “chassis(3)” or “module(9).”

Each row is indexed by an arbitrary, small integer, and contains a description and type of the physical entity. It also optionally contains the index number of another entPhysicalEntry indicating a containment relationship between the two.

Version 2 of the Entity MIB provides additional MIB objects for each physical entity. Some common read-only attributes have been added, as well as three writable string objects.

- entPhysicalAlias

  This string can be used by an NMS as a non-volatile identifier for the physical component. Maintaining a non-volatile string for every physical component represented in the entPhysicalTable can be costly and unnecessary. An agent may algorithmically generate 'entPhysicalAlias' strings for particular entries (for example, based on the entPhysicalClass value).

- entPhysicalAssetID

  This string is provided to store a user-specific asset identifier for removable physical components. In order to reduce the non-volatile storage needed by a particular agent, a network administrator should only assign asset identifiers to physical entities which are field-replaceable (that is, not permanently contained within another physical entity).

- entPhysicalSerialNum

  This string is provided to store a vendor-specific serial number string for physical components. This is a writable object in case an agent cannot identify the serial numbers of all installed physical entities, and a network administrator wishes to configure the non-volatile serial number strings manually (via an NMS application).

5.6.5.1.1 entPhysicalTable

This table contains one row per physical entity. There is always at least one row for an “overall” physical entity.

This table is a sequence of entPhysicalEntry which each entry provides objects (entPhysicalDescr, entPhysicalVendorType, and entPhysicalClass) to help an NMS identify and characterize the entry, and objects (entPhysicalContainedIn and entPhysicalParentRelPos) to help an NMS relate the particular entry to other entries in this table.

Please refer to Interface List, Section 11, page 29, for a list of supported ports and interfaces.

Index:

- entPhysicalIndex

OIDs:

- entPhysicalTable - {entityPhysical 1 } 1.3.6.1.2.1.47.1.1.1
- entPhysicalEntry - {entPhysicalTable 1 } 1.3.6.1.2.1.47.1.1.1.1

Table 47. entPhysicalTable

Object	OID	Syntax	Access	Status	Supported
entPhysicalIndex	{ entPhysicalEntry 1 }	PhysicalIndex	not-accessible	current	Y
entPhysicalDescr	{ entPhysicalEntry 2 }	SnmpAdminString	read-only	current	Y
entPhysicalVendorType	{ entPhysicalEntry 3 }	AutonomousType	read-only	current	Y
entPhysicalContainedIn	{ entPhysicalEntry 4 }	INTEGER(0…2147483647)	read-only	current	Y

Table 47. entPhysicalTable

Object	OID	Syntax	Access	Status	Supported
entPhysicalClass	{ entPhysicalEntry 5 }	PhysicalClass	read-only	current	Y
entPhysicalParentRelPos	{ entPhysicalEntry 6 }	INTEGER(-1…2147483647)	read-only	current	Y
entPhysicalName	{ entPhysicalEntry 7 }	SnmpAdminString	read-only	current	Y
entPhysicalHardwareRev	{ entPhysicalEntry 8 }	SnmpAdminString	read-only	current	Y
entPhysicalFirmwareRev	{ entPhysicalEntry 9 }	SnmpAdminString	read-only	current	N
entPhysicalSoftwareRev	{ entPhysicalEntry 10 }	SnmpAdminString	read-only	current	Y
entPhysicalSerialNum	{ entPhysicalEntry 11 }	SnmpAdminString(SIZE(0…32))	read-only[a]	current	Y
entPhysicalMfgName	{ entPhysicalEntry 12 }	SnmpAdminString	read-only	current	Y
entPhysicalModelName	{ entPhysicalEntry 13 }	SnmpAdminString	read-only	current	Y
entPhysicalAlias	{ entPhysicalEntry 14 }	SnmpAdminString(SIZE(0…32))	read-write	current	Y
entPhysicalAssetID	{ entPhysicalEntry 15 }	SnmpAdminString(SIZE(0…32))	read-write	current	Y
entPhysicalIsFRU	{ entPhysicalEntry 16 }	TruthValue	read-only	current	Y

a. Product imposed limit.

5.6.5.1.1.1 entPhysicalIndex

The index for this entry.

Valid Object Value(s)

The value of this object is based on the ifIndex in the Interface Group, Section 5.3.4.3.1, Page 37.

The entPhysicalIndex is encoded as a decimal number in the form of:

Where:

- R - Reserved bits. Implied as 0.
- CC - Chassis Number with the range of 01…99 and 00 is a special case to denote N/A
- SS - Slot Number with the range of 01…99 and 00 is a special case to denote N/A
  Slot Number is not applicable to this product and will always be 00
- L - Layer Number in the range of 1…9 and 0 is a special case to denote N/A
- PPP - Port/Interface Number in the range of 001…999 and 000 is a special case to denote N/A

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM, although considered “stackable,” are not managed in terms of a Single Managed Entity.

The following are values for the entPhyscialIndex object:

Table 48. entPhysicalIndex

Entity	Value										Comment
	R	**R**	**C**	**C**	**S**	**S**	**L**	**P**	**P**	**P**	
Chassis	0	0	0	1	0	0	0	0	0	0	
DSL Main Card	0	0	0	1	0	0	0	0	0	1	
MAUI Card	0	0	0	1	0	0	0	0	0	2	
POTS Splitter Child Card	0	0	0	1	0	0	0	0	0	3	
DSL Child Card	0	0	0	1	0	0	0	0	0	4	
Intake Temperature Sensor	0	0	0	1	0	0	0	0	5	1	
Internal Temperature Sensor	0	0	0	1	0	0	0	0	5	2	
Fan Speed Sensor *n*	0	0	0	1	0	0	0	0	6	*n*	*n* denotes the fan speed sensor number in the range: • 1…3 D/C powered chassis • 1…4 A/C powered chassis
Alarm Relay Input Contact	0	0	0	1	0	0	0	0	9	1	
Alarm Relay Output Major Contact	0	0	0	1	0	0	0	0	9	2	
Alarm Relay Output Minor Contact	0	0	0	1	0	0	0	0	9	3	
DSL Main Card PLD	0	0	0	1	0	0	0	1	0	1	
Reach DSL Main Card LED PLD[a]	0	0	0	1	0	0	0	1	0	2	
MAUI Card PLD	0	0	0	1	0	0	0	1	1	1	
DSL Child Card PLD	0	0	0	1	0	0	0	1	2	1	
Reach DSL Child Card LED PLD[a]	0	0	0	1	0	0	0	1	2	2	
Management Processor	0	0	0	1	0	0	0	2	0	1	
DSL Port *n*[b]	0	0	0	1	0	0	1	0	*n*	*n*	*nn* denotes the DSL port number in the range of 1…48
Console Port	0	0	0	1	0	0	1	1	0	1	
Ethernet Port *n*	0	0	0	1	0	0	1	2	0	*n*	*n* denotes the Ethernet port number in the range of 1…3[c]
T1/E1Port *n*[d]	0	0	0	1	0	0	1	4	0	*n*	*n* denotes the T1/E1 port number in the range of 1…8

a. ReachDSL Only

b. DSL Main Card contains 24 DSL Ports and an optional DSL Child Card contains an additional 24 DSL Ports

c. Ethernet Port 3 is Not available on units with T1/E1 ML PPP MAUI.

d. T1/E1 Ports are not available on units with GigE MAUI.

Note: Leading zeros may be omitted.

5.6.5.1.1.2 entPhysicalDescr

A textual description of physical entity. This object should contain a string which identifies the manufacturer's name for the physical entity, and should be set to a distinct value for each version or model of the physical entity.

Valid Object Value(s)

Table 49. entPhysicalDescr (1 of 5)

Entity	Value
Chassis (2600 Reach)	*Company Name*[a] *Family*[b] *Model Number*; *Model Description* where: *Model Number; Model Description* is: • *2600; 24 Reach DSL 2.2 Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 Reach DSL 2.2 Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 Reach DSL 2.2 Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 24 Reach DSL 2.2 Ports, Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 Reach DSL 2.2 Ports, Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 Reach DSL 2.2 Ports, Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 48 Reach DSL 2.2 Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 48 Reach DSL 2.2 Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 48 Reach DSL 2.2 Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* As determined by the Model Family Tree section in the *BitStorm Product Overview* document.

Table 49. entPhysicalDescr (2 of 5)

Entity	Value
Chassis (2600 ADSL)	*Company Name*[c] *Family*[d] *Model Number*; *Model Description* where: *Model Number; Model Description* is: • *2600; 24 ADSL Annex A Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 ADSL Annex A Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex A Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex B Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 ADSL Annex B Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex B Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex A Ports, Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 ADSL Annex A Ports, Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex A Ports, Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex B Ports, Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 ADSL Annex B Ports, Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 ADSL Annex B Ports, Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 48 ADSL Annex A Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 48 ADSL Annex A Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 48 ADSL Annex A Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* • *2600; 48 ADSL Annex B Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 48 ADSL Annex B Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 48 ADSL Annex B Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* As determined by the Model Family Tree section in the *BitStorm Product Overview* document.

Table 49. entPhysicalDescr (3 of 5)

Entity	Value
Chassis (2600 SHDSL)	*Company Name*[e] *Family*[f] *Model Number*; *Model Description* where: *Model Number; Model Description* is: • *2600; 24 SHDSL Ports, No Splitters, Universal AC, GigE uplink, stacking with 10/100 BT* • *2600; 24 SHDSL Ports, No Splitters, Universal AC, T1 ML PPP, stacking with 10/100 BT* • *2600; 24 SHDSL Ports, No Splitters, Universal AC, E1 ML PPP, stacking with 10/100 BT* As determined by the Model Family Tree section in the *BitStorm Product Overview* document.
Chassis (4200 Reach)	*Company Name*[g] *Family*[h] *Model Number*; *Model Description* where: *Model Number; Model Description* is: • *4200; 24 Reach DSL 2.2 Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 Reach DSL 2.2 Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 Reach DSL 2.2 Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 24 Reach DSL 2.2 Ports, Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 Reach DSL 2.2 Ports, Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 Reach DSL 2.2 Ports, Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 48 Reach DSL 2.2 Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 48 Reach DSL 2.2 Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 48 Reach DSL 2.2 Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* As determined by the Model Family Tree section in the *BitStorm Product Overview* document.

Table 49. entPhysicalDescr (4 of 5)

Entity	Value
Chassis (4200 ADSL)	*Company Name*[i] *Family*[j] *Model Number*; *Model Description* where: *Model Number; Model Description* is: • *4200; 24 ADSL Annex A Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 ADSL Annex A Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex A Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex B Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 ADSL Annex B Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex B Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex A Ports, Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 ADSL Annex A Ports, Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex A Ports, Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex B Ports, Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 ADSL Annex B Ports, Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 ADSL Annex B Ports, Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 48 ADSL Annex A Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 48 ADSL Annex A Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 48 ADSL Annex A Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* • *4200; 48 ADSL Annex B Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 48 ADSL Annex B Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 48 ADSL Annex B Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* As determined by the Model Family Tree section in the *BitStorm Product Overview* document.

Table 49. entPhysicalDescr (5 of 5)

Entity	Value
Chassis (4200 SHDSL)	*Company Name*[k] *Family*[l] *Model Number*; *Model Description* where: *Model Number; Model Description* is: • *4200; 24 SHDSL Ports, No Splitters, -48v DC, GigE uplink, stacking with 10/100 BT* • *4200; 24 SHDSL Ports, No Splitters, -48v DC, T1 ML PPP, stacking with 10/100 BT* • *4200; 24 SHDSL Ports, No Splitters, -48v DC, E1 ML PPP, stacking with 10/100 BT* As determined by the Model Family Tree section in the *BitStorm Product Overview* document.
DSL Main Card	*Company Name*[a] *Family*[b.] *Model Number*[m]; DSL Main Card
MAUI Card	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Management Aggregation Uplink Interface (MAUI) Card
POTS Splitter Child Card	*Company Name*[a] *Family*[b.] *Model Number*[m] ; POTS Splitter Child Card
DSL Child Card	*Company Name*[a] *Family*[b.] *Model Number*[m] ; DSL Child Card
Intake Temperature Sensor	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Intake Temperature Sensor
Internal Temperature Sensor	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Internal Temperature Sensor
Fan Speed Sensor *n*	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Fan Speed Sensor *n*
Alarm Relay Input Contact	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Alarm Relay Input Contact
Alarm Relay Output Major Contact	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Alarm Relay Output Major Contact
Alarm Relay Output Minor Contact	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Alarm Relay Output Minor Contact
DSL Main Card PLD	*Company Name*[a] *Family*[b.] *Model Number*[m] ; DSL Main Card PLD
Reach DSL Main Card LED PLD	*Company Name*[a] *Family*[b.] *Model Number*[m] ; DSL Main Card LED PLD
MAUI Card PLD	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Management Aggregation Uplink Interface (MAUI) Card PLD
DSL Child Card PLD	*Company Name*[a] *Family*[b.] *Model Number*[m] ; DSL Child Card PLD
Reach DSL Child Card LED PLD	*Company Name*[a] *Family*[b.] *Model Number*[m] ; DSL Child Card LED PLD
Management Processor	*Company Name*[a] *Family*[b.] *Model Number*[m] ; Management Processor
DSL Port *n*	*Company Name*[a] *Family*[b.] *Model Number*[m] ; DSL PORT *n* where *n* is 1…48
Console Port	*Company Name*[a] *Family*[b.] *Model Number*[m] ; CONSOLE PORT
Ethernet Port *n*	*Company Name*[a] *Family*[b.] *Model Number*[m] ; ETHERNET PORT *n*
T1/E1Port *n*	*Company Name*[a] *Family*[b.] *Model Number*[m] ; T1/E1 PORT *n*

a. Reference the sysDescr object in the MIB-II's System Group for clarification on *Company Name*.

b. The family is "BitStorm" for the 26xx products and "GranDSLAM" for the 42xx products.

c. Reference the sysDescr object in the MIB-II's System Group for clarification on *Company Name*.

d. The family is "BitStorm" for the 26xx products and "GranDSLAM" for the 42xx products.

e. Reference the sysDescr object in the MIB-II's System Group for clarification on *Company Name*.

f. The family is “BitStorm” for the 26xx products and “GranDSLAM” for the 42xx products.
g. Reference the sysDescr object in the MIB-II’s System Group for clarification on *Company Name*.
h. The family is “BitStorm” for the 26xx products and “GranDSLAM” for the 42xx products.
i. Reference the sysDescr object in the MIB-II’s System Group for clarification on *Company Name*.
j. The family is “BitStorm” for the 26xx products and “GranDSLAM” for the 42xx products.
k. Reference the sysDescr object in the MIB-II’s System Group for clarification on *Company Name*.
l. The family is “BitStorm” for the 26xx products and “GranDSLAM” for the 42xx products.
m. The Model Number is “2600” for the 26xx products and “4200” for the 42xx products.

5.6.5.1.1.3 entPhysicalVendorType

An indication of the vendor-specific hardware type of the physical entity. Note that this is different from the definition of MIB-II's sysObjectID.

An agent should set this object to a enterprise-specific registration identifier value indicating the specific equipment type in detail. The associated instance of entPhysicalClass is used to indicate the general type of hardware device.

If no vendor-specific registration identifier exists for this physical entity, or the value is unknown by this agent, then the value { 0 0 } is returned.

Valid Object Value(s)

Below is an OID tree for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. The table that follows describes the values supported for this object.

Table 50. OID Tree (1 of 2)

OID Prefix	OID	Object
Company Enterprise.[a]	1.14.17.2	ips-chassis
	1.14.17.2.2	ips-2611
	1.14.17.2.3	ips-2621
	1.14.17.2.4	ips-4219
	1.14.17.2.5	ips-4229
	1.14.17.2.6	ips-2671
	1.14.17.2.7	ips-4279
	1.14.17.2.9	ips-2631
	1.14.17.2.10	ips-4239
	1.14.17.2.11	ips-4214
	1.14.17.2.12	ips-4224
	1.14.17.2.13	ips-4234
	1.14.17.2.14	ips-4274
	1.14.17.2.15	ips-1614
	1.14.17.2.16	ips-2624
	1.14.17.2.17	ips-2634
	1.14.17.2.18	ips-2674
Company Enterprise.[a]	1.14.17.3	ips-fixed-cards
	1.14.17.3.1	ips-24port-adsl-main-card
	1.14.17.3.2	ips-24port-adsl-child-card
	1.14.17.3.3	ips-maui-card
	1.14.17.3.4	ips-pots-splitter-child-card
	1.14.17.3.5	ips-24port-reachDsl-main-card
	1.14.17.3.6	ips-24port-reachDsl-child-card
	1.14.17.3.7	ips-24port-shdsl-main-card
	1.14.17.3.8	ips-24port-shdsl-child-card
	1.14.17.3.9	ips-48port-adsl-main-card

Table 50. OID Tree (2 of 2)

OID Prefix	OID	Object
Company Enterprise.[a]	1.14.17.5	ips-ports
	1.14.17.5.2	ips-rs232-dce-port
	1.14.17.5.5	ips-ethernet-port
	1.14.17.5.6	ips-t1e1-port
	1.14.17.5.7	ips-adsl-a-port
	1.14.17.5.8	ips-reachDsl-port
	1.14.17.5.9	ips-shdsl-port
	1.14.17.5.10	ips-adsl-b-port
Company Enterprise.[a]	1.14.17.6	ips-components
	1.14.17.6.1	ips-fan
	1.14.17.6.2	ips-temperature-sensor
	1.14.17.6.3	ips-processor
	1.14.17.6.4	ips-pld
	1.14.17.6.5	ips-speed-sensor
	1.14.17.6.6	ips-alarm-relay-contact

a. Refer to sysObjectID, Section 5.2.3.1.2, page 22, for a definition of *Company Enterprise*.

Table 51. entPhysicalVendorType (1 of 2)

Entity	Value[a]
Chassis	One of: • { *Company Enterprise*[b].1.14.17.2.2 } ips-2611 • { *Company Enterprise*[b] .1.14.17.2.3 } ips-2621 • { *Company Enterprise*[b] .1.14.17.2.4 } ips-4219 • { *Company Enterprise*[b] .1.14.17.2.5 } ips-4229 • { *Company Enterprise*[b] .1.14.17.2.6 } ips-2671 • { *Company Enterprise*[b] .1.14.17.2.7 } ips-4279 • { *Company Enterprise*[b] .1.14.17.2.9 } ips-2631 • { *Company Enterprise*[b] .1.14.17.2.10 } ips-4239 • { *Company Enterprise*[b] .1.14.17.2.11 } ips-4214 • { *Company Enterprise*[b] .1.14.17.2.12 } ips-4224 • { *Company Enterprise*[b] .1.14.17.2.13 } ips-4234 • { *Company Enterprise*[b] .1.14.17.2.14 } ips-4274 • { *Company Enterprise*[b] .1.14.17.2.15 } ips-2614 • { *Company Enterprise*[b] .1.14.17.2.16 } ips-2624 • { *Company Enterprise*[b] .1.14.17.2.17 } ips-2634
DSL Main Card	One of: • { *Company Enterprise*[b] .1.14.17.3.1 } ips-24port-adsl-main-card • { *Company Enterprise*[b] .1.14.17.3.5 } ips-24port-reachDsl-main-card • { *Company Enterprise*[b] .1.14.17.3.7 } ips-24port-shdsl-main-card • { *Company Enterprise*[b] .1.14.17.3.9 } ips-48port-adsl-main-card
MAUI Card	{ *Company Enterprise*[b] .1.14.17.3.3 } ips-maui-card
POTS Splitter Child Card	{ *Company Enterprise*[b] .1.14.17.3.4 } ips-pots-splitter-child-card
DSL Child Card	One of: • { *Company Enterprise*[b] .1.14.17.3.2 } ips-24port-adsl-child-card • { *Company Enterprise*[b] .1.14.17.3.6 } ips-24port-reachDsl-child-card
Intake Temperature Sensor	{ *Company Enterprise*[b] .1.14.17.6.2 } ips-temperature-sensor
Internal Temperature Sensor	{ *Company Enterprise*[b] .1.14.17.6.2 } ips-temperature-sensor
Fan Speed Sensor *n*	{ *Company Enterprise*[b] .1.14.17.6.5 } ips-speed-sensor
Alarm Relay Input Contact	{ *Company Enterprise*[b] .1.14.17.6.6 } ips-alarm-relay-contact
Alarm Relay Output Major Contact	{ *Company Enterprise*[b] .1.14.17.6.6 } ips-alarm-relay-contact
Alarm Relay Output Minor Contact	{ *Company Enterprise*[b] .1.14.17.6.6 } ips-alarm-relay-contact
DSL Main Card PLD	{ *Company Enterprise*[b] .1.14.17.6.4 } ips-pld
Reach DSL Main Card LED PLD	{ *Company Enterprise*[b] .1.14.17.6.4 } ips-pld
MAUI PLD	{ *Company Enterprise*[b] .1.14.17.6.4 } ips-pld
DSL Child Card PLD	{ *Company Enterprise*[b] .1.14.17.6.4 } ips-pld

Table 51. entPhysicalVendorType (2 of 2)

Entity	Value[a]
Reach DSL Child Card LED PLD	{ *Company Enterprise*[b] .1.14.17.6.4 } ips-pld
Management Processor	{ *Company Enterprise*[b] .1.14.17.6.3 } ips-processor
DSL Port *n*	One of: • { *Company Enterprise*[b] .1.14.17.5.7 } ips-adsl-a-port • { *Company Enterprise*[b] .1.14.17.5.8 } ips-reachDsl-port • { *Company Enterprise*[b] .1.14.17.5.9 } ips-shdsl-port • { *Company Enterprise*[b] .1.14.17.5.10 } ips-adsl-b-port
Console Port	{ *Company Enterprise*[b] .1.14.17.5.2 } ips-rs232-dce-port
Ethernet Port *n*	{ *Company Enterprise*[b] .1.14.17.5.5 } ips-ethernet-port
T1/E1Port *n*	{ *Company Enterprise*[b] .1.14.17.5.6 } ips-t1e1-port

a. Note that some MIB browsers render the dash character '-' as an underscore character '_'.

b. Reference the sysObjectID object in MIB-II's System Group for clarification on *Company Enterprise*.

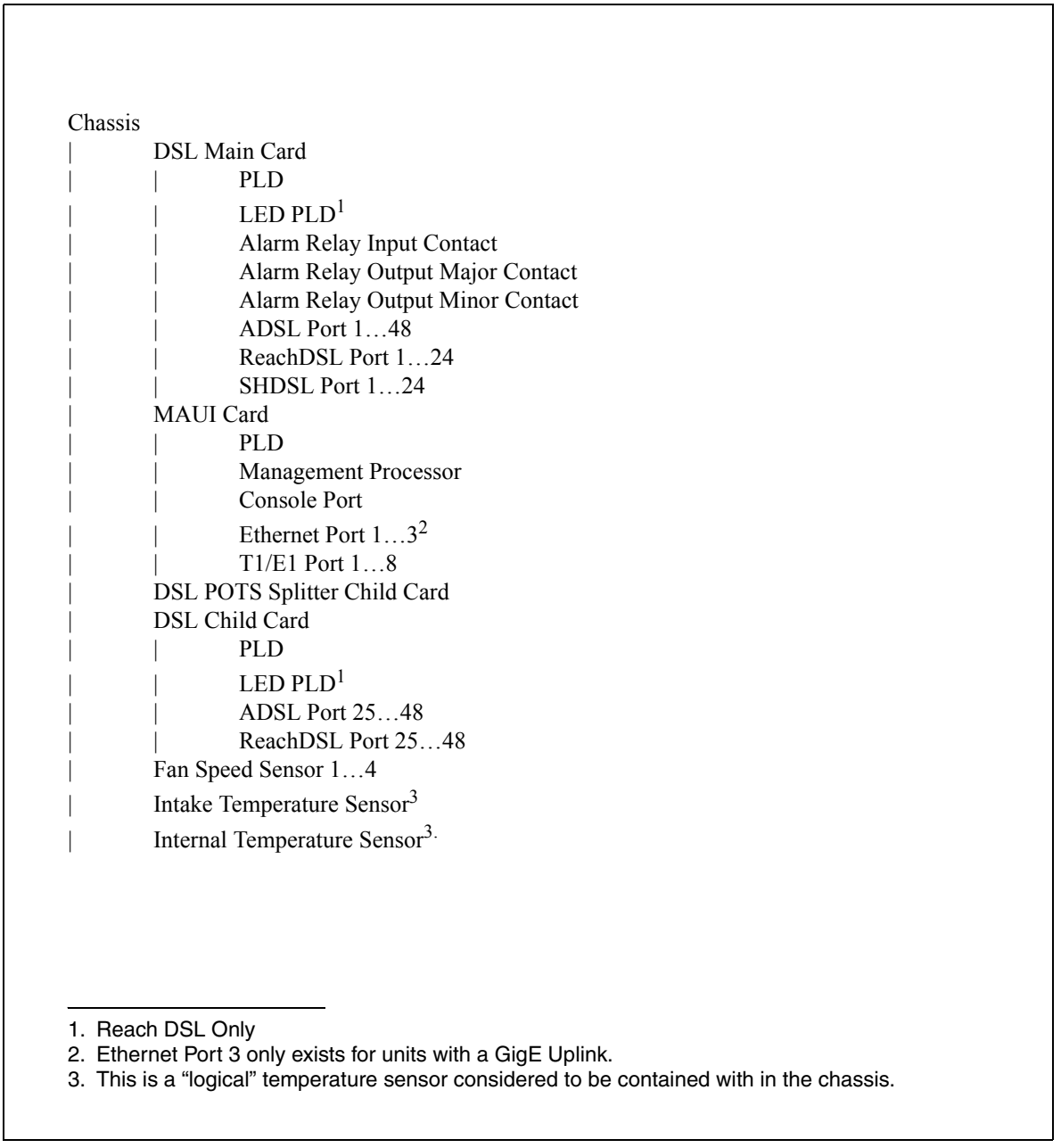
5.6.5.1.1.4 entPhysicalContainedIn

The value of entPhysicalIndex for the physical entity which "contains" this physical entity. A value of zero indicates this physical entity is not contained in any other physical entity. Note that the set of "containment" relationships define a strict hierarchy; that is, recursion is not allowed.

In the event a physical entity is contained by more than one physical entity (for example, double-wide modules), this object should identify the containing entity with the lowest value of entPhysicalIndex.

Valid Object Value(s)

A stack is not contained in anything. A chassis is contained in a stack. A chassis or a card contains a variety of entities. The Containment Tree, Figure , Page 100, illustrates the entities and their containment relationship.



Containment Tree

Table 52. entPhysicalContainedIn

Entity	Value
Chassis	0
DSL Main Card	entPhysicalIndex for the Chassis
MAUI Card	entPhysicalIndex for the Chassis
POTS Splitter Child Card	entPhysicalIndex for the Chassis
DSL Child Card	entPhysicalIndex for the Chassis
Intake Temperature Sensor[a]	entPhysicalIndex for the Chassis
Internal Temperature Sensor[a.]	entPhysicalIndex for the Chassis
Fan Speed Sensor *n*	entPhysicalIndex for the Chassis
Alarm Relay Input Contact	entPhysicalIndex for the DSL Main Card
Alarm Relay Output Major Contact	entPhysicalIndex for the DSL Main Card
Alarm Relay Output Minor Contact	entPhysicalIndex for the DSL Main Card
DSL Main Card PLD	entPhysicalIndex for the DSL Main Card
Reach DSL Main Card LED PLD	entPhysicalIndex for the DSL Main Card
MAUI PLD	entPhysicalIndex for the MAUI Card
DSL Child Card PLD	entPhysicalIndex for the DSL Child Card
Reach DSL Child Card LED PLD	entPhysicalIndex for the DSL Child Card
Management Processor	entPhysicalIndex for the MAUI Card
DSL Port *n*	One of: • entPhysicalIndex for the 24-port DSL Main Card, where *n* is 1…24 • entPhysicalIndex for the 24-port DSL Child Card, where *n* is 25…48 • entPhysicalIndex for the 48-port DSL Main Card, where n is 1…48
Console Port	entPhysicalIndex for the MAUI Card
Ethernet Port *n*	entPhysicalIndex for the MAUI Card
T1/E1Port *n*	entPhysicalIndex for the MAUI Card

a. This is a “logical” temperature sensor considered to be contained with in the chassis.

5.6.5.1.1.5 entPhysicalClass

An indication of the general hardware type of the physical entity.

An agent should set this object to the standard enumeration value which most accurately indicates the general class of the physical entity, or the primary class if there is more than one.

If no appropriate standard registration identifier exists for this physical entity, then the value “other(1)” is returned. If the value is unknown by this agent, then the value “unknown(2)” is returned.

Valid Object Value(s)

Table 53. entPhysicalClass

Entity	Value
Chassis	chassis(3)
DSL Main Card	module(9)
MAUI Card	module(9)
POTS Splitter Child Card	module(9)
DSL Child Card	module(9)
Intake Temperature Sensor	sensor(8)
Internal Temperature Sensor	sensor(8)
Fan Speed Sensor *n*	sensor(8)
Alarm Relay Input Contact	other(1)
Alarm Relay Output Major Contact	other(1)
Alarm Relay Output Minor Contact	other(1)
DSL Main Card PLD	other(1)
Reach DSL Main Card LED PLD	other(1)
MAUI PLD	other(1)
DSL Child Card PLD	other(1)
Reach DSL Child Card LED PLD	other(1)
Management Processor	other(1)
DSL Port *n*	port(10)
Console Port	port(10)
Ethernet Port *n*	port(10)
T1/E1Port *n*	port(10)

5.6.5.1.1.6 entPhysicalParentRelPos

An indication of the relative position of this “child” component among all its “sibling” components. Sibling components are defined as entPhysicalEntries —entries in the entPhysicalTable— which share the same instance values of each of the entPhysicalContainedIn and entPhysicalClass objects.

An NMS can use this object to identify the relative ordering for all sibling components of a particular parent (identified by the entPhysicalContainedIn instance in each sibling entry).

This value should match any external labeling of the physical component if possible. For example, for a module labeled as “card #3,” entPhysicalParentRelPos should have the value “3.” Note that the entPhysicalEntry for the module plugged in slot 3 should have an entPhysicalParentRelPos value of “1.”

If the physical position of this component does not match any external numbering or clearly visible ordering, then user documentation or other external reference material should be used to determine the parent-relative position. If this is

not possible, then the agent should assign a consistent (but possibly arbitrary) ordering to a given set of “sibling” components, perhaps based on internal representation of the components.

If the agent cannot determine the parent-relative position for some reason, or if the associated value of entPhysicalContainedIn is “0,” then the value “-1” is returned. Otherwise a non-negative integer is returned, indicating the parent-relative position of this physical entity.

Parent-relative ordering normally starts from “1” and continues to “N,” where “N” represents the highest positioned child entity. However, if the physical entities (for example, slots) are labeled from a starting position of zero, then the first sibling should be associated with a entPhysicalParentRelPos value of “0.” Note that this ordering may be sparse or dense, depending on agent implementation.

The actual values returned are not globally meaningful, as each “parent” component may use different numbering algorithms. The ordering is only meaningful among siblings of the same parent component.

The agent should retain parent-relative position values across reboots, either through algorithmic assignment or use of non-volatile storage.

Valid Object Value(s)

Table 54. EntPhysicalParentRelPos

Entity	Index	Contained In		Class	Value
		Entity	Entity's Index		
Chassis	1 000 000	Nothing	00 000 000	chassis(3)	**-1**
DSL Main Card	1 000 001	Chassis	1 000 000	module(9)	**1**
MAUI Card	1 000 002	Chassis	1 000 000	module(9)	**2**
POTS Splitter Child Card	1 000 003	Chassis	1 000 000	module(9)	**3**
DSL Child Card	1 000 004	Chassis	1 000 000	module(9)	**3**
Intake Temperature Sensor	1 000 051	Chassis	1 000 000	sensor(8)	**1**
Internal Temperature Sensor	1 000 052	Chassis	1 000 000	sensor(8)	**2**
Fan Speed Sensor *n*	1 000 06*n*	Chassis	1 000 000	sensor(8)	***3…6***
Alarm Relay Input Contact	1 000 091	DSL Main Card	1 000 001	other(1)	**1**
Alarm Relay Output Major Contact	1 000 092	DSL Main Card	1 000 001	other(1)	**2**
Alarm Relay Output Minor Contact	1 000 093	DSL Main Card	1 000 001	other(1)	**3**
DSL Main Card PLD	1 000 101	DSL Main Card	1 000 001	other(1)	**4**
Reach DSL Main Card LED PLD	1 000 102	DSL Main Card	1 000 001	other(1)	**5**
MAUI PLD	1 000 111	MAUI Card	1 000 002	other(1)	**1**
DSL Child Card PLD	1 000 121	DSL Child Card	1 000 004	other(1)	**1**
Reach DSL Child Card LED PLD	1 000 122	DSL Child Card	1 000 004	other(1)	**2**
Management Processor	1 000 201	MAUI Card	1 000 002	other(1)	**2**
DSL Port *n* = 1…24	1 001 0*nn*	DSL Main Card	1 000 001	port(10)	**1…24**
DSL Port *n* = 25…48		DSL Child Card	1 000 001	port(10)	**1…24**
Console Port	1 001 101	MAUI Card	1 000 002	port(10)	**1**
Ethernet Port *n=1…3*	1 001 201	MAUI Card	1 000 002	port(10)	**2…4**
T1/E1Port *n*	1 001 90*n*	MAUI Card	1 000 002	port(10)	**4…11**

5.6.5.1.1.7 entPhysicalName

The textual name of the physical entity. The value of this object should be the name of the component 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 "console" or a simple component number (for example, port or module number), such as "1," depending on the physical component naming syntax of the device.

If there is no local name, or this object is otherwise not applicable, then this object contains a zero-length string.

Note that the value of entPhysicalName for two physical entities is the same in the event that the console interface does not distinguish between them: for example, slot-1 and the card in slot-1.

Valid Object Value(s)

Table 55. entPhysicalName

Entity	Value
Chassis	One of: • BitStorm 2600 ADSL Chassis • BitStorm 2600 ReachDSL Chassis • BitStorm 2600 SHDSL Chassis • GranDSLAM 4200 ADSL Chassis • GranDSLAM 4200 ReachDSL Chassis • GranDSLAM 4200 SHDSL Chassis
DSL Main Card	One of: • ADSL Main Card • ReachDSL Main Card • SHDSL Main Card
MAUI Card	Management Aggregation Uplink Interface (MAUI) Card
POTS Splitter Child Card	POTS Splitter Child Card
DSL Child Card	One of: • ADSL Child Card • ReachDSL Child Card
Intake Temperature Sensor	Intake Temperature Sensor
Internal Temperature Sensor	Internal Temperature Sensor
Fan Speed Sensor *n*	Fan Speed Sensor *n*
Alarm Relay Input Contact	Alarm Relay Input Contact
Alarm Relay Output Major Contact	Alarm Relay Output Major Contact
Alarm Relay Output Minor Contact	Alarm Relay Output Minor Contact
DSL Main Card PLD	DSL Main Card PLD
Reach DSL Main Card LED PLD	Reach DSL Main Card LED PLD
MAUI PLD	Management Aggregation Uplink Interface (MAUI) Card PLD
DSL Child Card PLD	DSL Child Card PLD
Reach DSL Child Card LED PLD	Reach DSL Child Card LED PLD
Management Processor	Management Processor
DSL Port *n*	dsl *n*
Console Port	console
Ethernet Port *n*	ethernet *n*
T1/E1Port *n*	t1/e1 *n*

5.6.5.1.1.8 entPhysicalHardwareRev

The vendor-specific hardware revision string for the physical entity. The preferred value is the hardware revision identifier actually printed on the component itself (if present).

Note that if revision information is stored internally in a non-printable (for example, binary) format, then the agent must convert such information to a printable format, in an implementation-specific manner.

If no specific hardware revision string is associated with the physical component, or this information is unknown to the agent, then this object will contain a zero-length string.

Valid Object Value(s)

Table 56. entPhysicalHardwareRev

Entity	Value
Chassis	String appropriate for representing the revision of this entity
DSL Main Card	String appropriate for representing the revision of this entity
MAUI Card	String appropriate for representing the revision of this entity
POTS Splitter Child Card	String appropriate for representing the revision of this entity
DSL Child Card	String appropriate for representing the revision of this entity
Intake Temperature Sensor	zero-length string
Internal Temperature Sensor	zero-length string
Fan Speed Sensor *n*	zero-length string
Alarm Relay Input Contact	zero-length string
Alarm Relay Output Major Contact	zero-length string
Alarm Relay Output Minor Contact	zero-length string
DSL Main Card PLD	String appropriate for representing the revision of this entity
Reach DSL Main Card LED PLD	String appropriate for representing the revision of this entity
MAUI PLD	String appropriate for representing the revision of this entity
DSL Child Card PLD	String appropriate for representing the revision of this entity
Reach DSL Child Card LED PLD	String appropriate for representing the revision of this entity
Management Processor	zero-length string
DSL Port *n*	zero-length string
Console Port	zero-length string
Ethernet Port *n*	zero-length string
T1/E1Port *n*	zero-length string

5.6.5.1.1.9 entPhysicalFirmwareRev

The vendor-specific firmware revision string for the physical entity.

Note that if revision information is stored internally in a non-printable (for example, binary) format, then the agent must convert such information to a printable format, in an implementation-specific manner.

If no specific firmware programs are associated with the physical component, or this information is unknown to the agent, then this object will contain a zero-length string.

Valid Object Value(s)

This object is not supported.

5.6.5.1.1.10 entPhysicalSoftwareRev

The vendor-specific software revision string for the physical entity.

Note that if revision information is stored internally in a non-printable (for example, binary) format, then the agent must convert such information to a printable format, in an implementation-specific manner.

If no specific software programs are associated with the physical component, or this information is unknown to the agent, then this object will contain a zero-length string.

Valid Object Value(s)

Since the operation of the BitStorm 2600 and GranDSLAM 4200 IP DSLAM is such that there is one software revision per chassis, this object returns a zero-length string for all entities except the chassis. The chassis will return the revision as a string in the form of yy.yy.yy.

5.6.5.1.1.11 entPhysicalSerialNum

The vendor-specific serial number string for the physical entity. The preferred value is the serial number string actually printed on the component itself (if present).

On the first instantiation of an physical entity, the value of entPhysicalSerialNum associated with that entity is set to the correct vendor-assigned serial number, if this information is available to the agent. If a serial number is unknown or non-existent, the entPhysicalSerialNum is set to a zero-length string instead.

Note that implementations which can correctly identify the serial numbers of all installed physical entities do not need to provide write access to the entPhysicalSerialNum object. Agents which cannot provide non-volatile storage for the entPhysicalSerialNum strings are not required to implement write access for this object.

Not every physical component will have a serial number, or even need one. Physical entities for which the associated value of the entPhysicalIsFRU object is equal to “false(2)” (for example, the repeater ports within a repeater module), do not need their own unique serial number. An agent does not have to provide write access for such entities, and may return a zero-length string.

If write access is implemented for an instance of entPhysicalSerialNum, and a value is written into the instance, the agent must retain the supplied value in the entPhysicalSerialNum instance associated with the same physical entity for as long as that entity remains instantiated. This includes instantiations across all re- initializations/reboots of the network management system, including those which result in a change of the physical entity's entPhysicalIndex value.

Valid Object Value(s)

Table 57. entPhysicalSerialNum

Entity	Value
Chassis	String appropriate for representing the serial number of this entity
DSL Main Card	String appropriate for representing the serial number of this entity
MAUI Card	String appropriate for representing the serial number of this entity
POTS Splitter Child Card	String appropriate for representing the serial number of this entity
DSL Child Card	String appropriate for representing the serial number of this entity
Intake Temperature Sensor	zero-length string
Internal Temperature Sensor	zero-length string
Fan Speed Sensor *n*	zero-length string
Alarm Relay Input Contact	zero-length string
Alarm Relay Output Major Contact	zero-length string
Alarm Relay Output Minor Contact	zero-length string
DSL Main Card PLD	zero-length string
Reach DSL Main Card LED PLD	zero-length string
MAUI PLD	zero-length string
DSL Child Card PLD	zero-length string
Reach DSL Child Card LED PLD	zero-length string
Management Processor	zero-length string
DSL Port *n*	zero-length string
Console Port	zero-length string
Ethernet Port *n*	zero-length string
T1/E1Port *n*	zero-length string

5.6.5.1.1.12 entPhysicalMfgName

The name of the manufacturer of this physical component. The preferred value is the manufacturer name string actually printed on the component itself (if present).

Note that comparisons between instances of the entPhysicalModelName, entPhysicalFirmwareRev, entPhysicalSoftwareRev, and the entPhysicalSerialNum objects, are only meaningful amongst entPhysicalEntries with the same value of entPhysicalMfgName.

If the manufacturer name string associated with the physical component is unknown to the agent, then this object will contain a zero-length string.

Valid Object Value(s)

The value of this object is “Paradyne.” for all entities.

5.6.5.1.1.13 entPhysicalModelName

The vendor-specific model name identifier string associated with this physical component. The preferred value is the customer-visible part number, which may be printed on the component itself.

If the model name string associated with the physical component is unknown to the agent, then this object will contain a zero-length string.

Valid Object Value(s)

Table 58. entPhysicalModelName

Entity	Value
Chassis	String appropriate for representing the part number of this entity
DSL Main Card	String appropriate for representing the part number of this entity
MAUI Card	String appropriate for representing the part number of this entity
POTS Splitter Child Card	String appropriate for representing the part number of this entity
DSL Child Card	String appropriate for representing the part number of this entity
Intake Temperature Sensor	zero-length string
Internal Temperature Sensor	zero-length string
Fan Speed Sensor *n*	zero-length string
Alarm Relay Input Contact	zero-length string
Alarm Relay Output Major Contact	zero-length string
Alarm Relay Output Minor Contact	zero-length string
DSL Main Card PLD	zero-length string
Reach DSL Main Card LED PLD	zero-length string
MAUI PLD	zero-length string
DSL Child Card PLD	zero-length string
Reach DSL Child Card LED PLD	zero-length string
Management Processor	zero-length string
DSL Port *n*	zero-length string
Console Port	zero-length string
Ethernet Port *n*	zero-length string
T1/E1Port *n*	zero-length string

5.6.5.1.1.14 entPhysicalAlias

This object is an “alias” name for the physical entity as specified by a network manager, and provides a non-volatile “handle” for the physical entity.

On the first instantiation of an physical entity, the value of entPhysicalAlias associated with that entity is set to the zero-length string. However, agent may set the value to a locally unique default value, instead of a zero-length string.

If write access is implemented for an instance of entPhysicalAlias, and a value is written into the instance, the agent must retain the supplied value in the entPhysicalAlias instance associated with the same physical entity for as long as that entity remains instantiated. This includes instantiations across all re- initializations/reboots of the network management system, including those which result in a change of the physical entity's entPhysicalIndex value.

Valid Object Value(s)

This object will default to a zero-length string.

5.6.5.1.1.15 entPhysicalAssetID

This object is a user-assigned asset tracking identifier for the physical entity as specified by a network manager, and provides non-volatile storage of this information.

On the first instantiation of an physical entity, the value of entPhysicalAssetID associated with that entity is set to the zero-length string.

Not every physical component will have a asset tracking identifier, or even need one. Physical entities for which the associated value of the entPhysicalIsFRU object is equal to “false(2)” (for example, the repeater ports within a repeater module), do not need their own unique asset tracking identifier. An agent does not have to provide write access for such entities, and may instead return a zero-length string.

If write access is implemented for an instance of entPhysicalAssetID, and a value is written into the instance, the agent must retain the supplied value in the entPhysicalAssetID instance associated with the same physical entity for as long as that entity remains instantiated. This includes instantiations across all re- initializations/reboots of the network management system, including those which result in a change of the physical entity's entPhysicalIndex value.

If no asset tracking information is associated with the physical component, then this object will contain a zero- length string.

Valid Object Value(s)

This object will default to a zero-length string.

5.6.5.1.1.16 entPhysicalIsFRU

This object indicates whether or not this physical entity is considered a “field replaceable unit” by the vendor. If this object contains the value “true(1)” then this entPhysicalEntry identifies a field replaceable unit. For all entPhysicalEntries which represent components that are permanently contained within a field replaceable unit, the value “false(2)” should be returned for this object.

Valid Object Value(s)

This object will take the value of true(1) for the following entities:

- Chassis

The value for all other entities is false(2).

5.6.5.2 entityLogical Group

This group contains a single table to identify logical entities, called the entLogicalTable.

The entLogicalTable contains one row per logical entity. Each row is indexed by an arbitrary, small integer and contains a name, description, and type of the logical entity.

Valid Object Value(s)

This group is not supported as there is only entity.

5.6.5.2.1 entLogicalTable

This table contains one row per logical entity. For agents which implement more than one naming scope, at least one entry must exist. Agents which instantiate all MIB objects within a single naming scope are not required to implement this table.

This table is a sequence of entLogicalEntry which contains information about a particular logical entity. Entities may be managed by this agent or other SNMP agents (possibly) in the same chassis.

Index:

- entLogicalIndex

OIDs:

- entLogicalTable - {entityLogical 1 } 1.3.6.1.2.1.47.1.2.1
- entLogicalEntry - {entLogicalTable 1 } 1.3.6.1.2.1.47.1.2.1.1

Table 59. entLogicalTable

Object	OID	Syntax	Access	Status	Supported
entLogicalIndex	{ entLogicalEntry 1 }	INTEGER(1…2147483647)	not-accessible	current	N
entLogicalDescr	{ entLogicalEntry 2 }	DisplayString	read-only	current	N
entLogicalType	{ entLogicalEntry 3 }	AutonomousType	read-only	current	N
entLogicalCommunity	{ entLogicalEntry 4 }	OCTET STRING (SIZE (1…255))	read-only	current	N
entLogicalTAddress	{ entLogicalEntry 5 }	TAddress	read-only	current	N
entLogicalTDomain	{ entLogicalEntry 6 }	TDomain	read-only	current	N
entLogicalContextEngineID	{ entLogicalEntry 7 }	SnmpEngineIdOrNone	read-only	current	N
entLoticalContextName	{ entLogicalEntry 8 }	SnmpAdminString	read-only	current	N

5.6.5.2.2 entLogicalIndex

The value of this object uniquely identifies the logical entity. The value is a small positive integer; index values for different logical entities are not necessarily contiguous.

5.6.5.2.3 entLogicalDescr

A textual description of the logical entity. This object should contain a string which identifies the manufacturer's name for the logical entity, and should be set to a distinct value for each version of the logical entity.

5.6.5.2.4 entLogicalType

An indication of the type of logical entity. This will typically be the OBJECT IDENTIFIER name of the node in the SMI's naming hierarchy which represents the major MIB module, or the majority of the MIB modules, supported by the logical entity. For example:

- a logical entity of a regular host/router -> mib-2
- a logical entity of a 802.1d bridge -> dot1dBridge
- a logical entity of a 802.3 repeater -> snmpDot3RptrMgmt

If an appropriate node in the SMI's naming hierarchy cannot be identified, the value “mib-2” should be used.

5.6.5.2.5 entLogicalCommunity

An SNMPv1 or SNMPv2C community-string which can be used to access detailed management information for this logical entity. The agent should allow read access with this community string (to an appropriate subset of all managed objects) and may also return a community string based on the privileges of the request used to read this object. Note that an agent may return a community string with read-only privileges, even if this object is accessed with a read-write community string. However, the agent must take care not to return a community string which allows more privileges than the community string used to access this object.

A compliant SNMP agent may wish to conserve naming scopes by representing multiple logical entities in a single “default” naming scope. This is possible when the logical entities represented by the same value of

entLogicalCommunity have no object instances in common. For example, “bridge1” and “repeater1” may be part of the main naming scope, but at least one additional community string is needed to represent “bridge2” and “repeater2.”

Logical entities “bridge1” and “repeater1” would be represented by sysOREntries associated with the “default” naming scope.

For agents not accessible via SNMPv1 or SNMPv2C, the value of this object is the empty string. This object may also contain an empty string if a community string has not yet been assigned by the agent, or no community string with suitable access rights can be returned for a particular SNMP request.

Note that this object is deprecated. Agents which implement SNMPv3 access should use the entLogicalContextEngineID and entLogicalContextName objects to identify the context associated with each logical entity. SNMPv3 agents may return a zero-length string for this object, or may continue to return a community string (for example, tri-lingual agent support).

5.6.5.2.6 entLogicalTAddress

The transport service address by which the logical entity receives network management traffic, formatted according to the corresponding value of entLogicalTDomain.

For snmpUDPDomain, a TAddress is 6 octets long, the initial 4 octets containing the IP-address in network-byte order and the last 2 containing the UDP port in network-byte order. Consult “Transport Mappings for Version 2 of the Simple Network Management Protocol” (RFC 1906) for further information on snmpUDPDomain.

5.6.5.2.7 entLogicalTDomain

Indicates the kind of transport service by which the logical entity receives network management traffic. Possible values for this object are presently found in the Transport Mappings for SNMPv2 document (RFC 1906).

5.6.5.2.8 entLogicalContextEngineID

The authoritative contextEngineID that can be used to send an SNMP message concerning information held by this logical entity, to the address specified by the associated “entLogicalTAddress/entLogicalTDomain” pair.

This object, together with the associated entLogicalContextName object, defines the context associated with a particular logical entity, and allows access to SNMP engines identified by a contextEngineId and contextName pair.

If no value has been configured by the agent, a zero-length string is returned, or the agent may choose not to instantiate this object at all.

5.6.5.2.9 entLogicalContextName

“The contextName that can be used to send an SNMP message concerning information held by this logical entity, to the address specified by the associated “entLogicalTAddress/entLogicalTDomain” pair.

This object, together with the associated entLogicalContextEngineID object, defines the context associated with a particular logical entity, and allows access to SNMP engines identified by a contextEngineId and contextName pair.

If no value has been configured by the agent, a zero-length string is returned, or the agent may choose not to instantiate this object at all.

5.6.5.3 entityMapping Group

This group contains three tables to identify associations between different system components.

The entLPMappingTable contains mappings between entLogicalIndex values (logical entities) and entPhysicalIndex values (the physical components supporting that entity). A logical entity can map to more than one physical component, and more than one logical entity can map to (share) the same physical component. If an agent represents a single logical entity, or multiple logical entities within a single naming scope, then implementation of this table may be omitted by the agent.

The entAliasMappingTable contains mappings between entLogicalIndex, entPhysicalIndex pairs and “alias” object identifier values. This allows resources managed with other MIBs (for example, repeater ports, bridge ports, physical and logical interfaces) to be identified in the physical entity hierarchy. Note that each alias identifier is only relevant in a particular naming scope. If an agent represents a single logical entity, or multiple logical entities within a single naming scope, then implementation of this table may be omitted by the agent.

The entPhysicalContainsTable contains simple mappings between “entPhysicalContainedIn” values for each container/”containee” relationship in the managed system. The indexing of this table allows an NMS to quickly discover the “entPhysicalIndex” values for all children of a given physical entity.

5.6.5.3.1 entLPMappingTable

This table contains zero or more rows of logical entity to physical equipment associations. For each logical entity known by this agent, there are zero or more mappings to the physical resources which are used to realize that logical entity.

An agent should limit the number and nature of entries in this table such that only meaningful and non-redundant information is returned. For example, in a system which contains a single power supply, mappings between logical entities and the power supply are not useful and should not be included.

Also, only the most appropriate physical component which is closest to the root of a particular containment tree should be identified in an entLPMapping entry.

For example, suppose a bridge is realized on a particular module, and all ports on that module are ports on this bridge. A mapping between the bridge and the module would be useful, but additional mappings between the bridge and each of the ports on that module would be redundant (since the entPhysicalContainedIn hierarchy can provide the same information). If, on the other hand, more than one bridge was utilizing ports on this module, then mappings between each bridge and the ports it used would be appropriate.

Also, in the case of a single backplane repeater, a mapping for the backplane to the single repeater entity is not necessary.

This table is a sequence of entLPMappingEntry which consists of information about a particular logical entity to physical equipment association. Note that the nature of the association is not specifically identified in this entry. It is expected that sufficient information exists in the MIBs used to manage a particular logical entity to infer how physical component information is utilized.

Indices:

- entLogicalIndex (Section 5.6.5.2.2, Page 111)
- entLPPyhsicalIndex

OIDs:

- entLPMappingTable - { entityMapping 1 } 1.3.6.1.2.1.47.1.3.1
- entLPMappingEntry - { entLPMappingTable 1 } 1.3.6.1.2.1.47.1.3.1.1

Table 60. entLPMappingTable

Object	OID	Syntax	Access	Status	Supported
entLPPhysicalIndex	{ entLPMappingEntry 1 }	PhysicalIndex	not--accessible	current	N

5.6.5.3.2 entLPPhysicalIndex

The value of this object identifies the index value of a particular entPhysicalEntry associated with the indicated entLogicalEntity.

5.6.5.4 entAliasMappingTable

This logical entity/component to alias table contains zero or more rows, representing mappings of logical entity and physical component to external MIB identifiers. Each physical port in the system may be associated with a mapping to an external identifier, which itself is associated with a particular logical entity's naming scope. A “wildcard” mechanism is provided to indicate that an identifier is associated with more than one logical entity.

This table is a sequence of entAliasMappingEntry which consists of information about a particular physical equipment, logical entity to external identifier binding. Each logical entity/physical component pair may be associated

with one alias mapping. The logical entity index may also be used as a “wildcard” (refer to the entAliasLogicalIndexOrZero object DESCRIPTION clause for details.)

Note that only entPhysicalIndex values which represent physical ports (that is, associated entPhysicalClass value is “port(10)”) are permitted to exist in this table.

Indices:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)
- entAliasLogicalIndexOrZero

OIDs:

- entAliasMappingTable - { entityMapping 2 } 1.3.6.1.2.1.47.1.3.2
- entAliasMappingEntry - { entAliasMappingTable 1 } 1.3.6.1.2.1.47.1.3.2.1

Table 61. entAliasMappingTable

Object	OID	Syntax	Access	Status	Supported
entAliasLogicalIndexOrZero	{ entAliasMappingEntry 1 }	INTEGER (0…2147483647)	not--accessible	current	Y
entAliasMappingIdentifier	{ entAliasMappingEntry 2 }	RowPointer	read-only	current	Y

5.6.5.4.1 entAliasLogicalIndexOrZero

The value of this object uniquely identifies the logical entity which defines the naming scope for the associated instance of the “entAliasMappingIdentifier” object.

If this object has a non-zero value, then it identifies the logical entity named by the same value of entLogicalIndex.

If this object has a value of zero, then the mapping between the physical component and the alias identifier for this entAliasMapping entry is associated with all unspecified logical entities. That is, a value of zero (the default mapping) identifies any logical entity which does not have an explicit entry in this table for a particular entPhysicalIndex/ entAliasMappingIdentifier pair.

For example, to indicate that a particular interface (for example, physical component 33) is identified by the same value of ifIndex for all logical entities, the following instance might exist:

- entAliasMappingIdentifier.33.0 = ifIndex.5

In the event an entPhysicalEntry is associated differently for some logical entities, additional entAliasMapping entries may exist, for example:

- entAliasMappingIdentifier.33.0 = ifIndex.6
- entAliasMappingIdentifier.33.4 = ifIndex.1
- entAliasMappingIdentifier.33.5 = ifIndex.1
- entAliasMappingIdentifier.33.10 = ifIndex.12

Note that entries with non-zero entAliasLogicalIndexOrZero index values have precedence over any zero-indexed entry. In this example, all logical entities except 4, 5, and 10, associate physical entity 33 with ifIndex.6.

Valid Object Value(s)

Since this object refers to a logical entity and there is only one logical entity in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM product, this object will always be 0.

5.6.5.4.2 entAliasMappingIdentifier

The value of this object identifies a particular conceptual row associated with the indicated entPhysicalIndex and entLogicalIndex pair.

Since only physical ports are modeled in this table, only entries which represent interfaces or ports are allowed. If an ifEntry exists on behalf of a particular physical port, then this object should identify the associated “ifEntry,” For repeater ports, the appropriate row in the “rptrPortGroupTable” should be identified instead.

For example, suppose a physical port was represented by entPhysicalEntry.3, entLogicalEntry.15 existed for a repeater, and entLogicalEntry.22 existed for a bridge. Then there might be two related instances of entAliasMappingIdentifier:

- entAliasMappingIdentifier.3.15 == rptrPortGroupIndex.5.2
- entAliasMappingIdentifier.3.22 == ifIndex.17

It is possible that other mappings (besides interfaces and repeater ports) may be defined in the future, as required.

Bridge ports are identified by examining the Bridge MIB and appropriate ifEntries associated with each “dot1dBasePort,” and are thus not represented in this table.

Valid Object Value(s)

This object contains the ifIndex associated with the particular port. Note that the index for this table is the entAliasLogicalIndexOrZero object which is always 0 and the entPhysicalIndex object which is the ifIndex.

5.6.5.5 entPhysicalContainsTable

The physical mapping table exposes the container/”containee” relationships between physical entities. This table provides equivalent information found by constructing the virtual containment tree for a given entPhysicalTable but in a more direct format.

In the event a physical entity is contained by more than one other physical entity (for example, double-wide modules), this table should include these additional mappings, which cannot be represented in the entPhysicalTable virtual containment tree.

This table is a sequence of entPhysicalContainsEntry which contains a single container/”containee” relationship.

Indices:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)
- entPhysicalChildIndex

OIDs:

- entPhysicalContainsTable - { entityMapping 3 } 1.3.6.1.2.1.47.1.3.3
- entPhysicalContainsEntry - { entPhysicalContainsTable 1 } 1.3.6.1.2.1.47.1.3.3.1

Table 62. entPhysicalContainsTable

Object	OID	Syntax	Access	Status	Supported
entPhysicalChildIndex	{ entPhysicalContainsEntry 1 }	PhysicalIndex	not-accessible	current	Y

5.6.5.5.1 entPhysicalChildIndex

The value of entPhysicalIndex (Section 5.6.5.1.1.1, Page 87) for the contained physical entity.

Valid Object Value(s)

The Containment Tree is diagramed in Figure , Page 100. This object constructs a table in the following form:

Table 63. entPhysicalChildIndex[a]

Entity	entPhysicalIndex	entPhysicalChildIndex	Comments
Chassis	1 000 000	1 000 001	DSL Main Card
		1 000 002	MAUI Card
		1 000 003	POTS Splitter Child Card
		1 000 004	DSL Child Card
		1 000 051	Intake Temperature Sensor
		1 000 052	Internal Temperature Sensor
		1 000 06*n*	A chassis contains up to 4 Fan Speed Sensors, where *n* is 1…4
DSL Main Card	1 000 001	1 000 091	Alarm Relay Input Contact
		1 000 092	Alarm Relay Output Major Contact
		1 000 093	Alarm Relay Output Minor Contact
		1 000 101	DSL Main Card PLD
		1 000 102	Reach DSL Main Card LED PLD
		1 001 *nnn*	ADSL, ReachDSL, or SHDSL Ports, where *nnn* is 1…24
MAUI Card	1 000 002	1 000 111	MAUI Card PLD
		1 000 201	Management Processor
		1 001 101	Console Port
		1 001 20*n*	Ethernet Ports, where *n* is 1 …3
		1 001 40*n*	T1/E1 Ports, where *n* is 1 …8
DSL Child Card	1 000 004	1 000 121	DSL Child Card PLD
		1 000 122	Reach DSL Child Card LED PLD
		1 001 *0nn*	ADSL or ReachDSL Ports, where *nn* is 25…48

a. Spaces added to increase readability. They do not actually exist as part of the value.
xx represents the chassis/slice/unit/box number.

5.6.5.6 entityGeneral Group

This group contains general information relating to the other object groups.

At this time, the entGeneral group contains a single scalar object (entLastChangeTime), which represents the value of sysUptime when any part of the system configuration last changed.

OIDs:

- entityGeneral - { entityMIBObjects 4 } 1.3.6.1.2.1.47.1.4

Table 64. entLastChangeTime

Object	OID	Syntax	Access	Status	Supported
entLastChangeTime	{ entityGeneral 1 }	TimeStamp	read-only	current	Y

5.6.5.6.1 entLastChangeTime

The value of sysUpTime at the time any of these events occur:

- a conceptual row is created, modified, or deleted in any of these tables:
  - — entPhysicalTable
  - — entLogicalTable
  - — entLPMappingTable
  - — entAliasMappingTable
  - — entPhysicalContainsTable

Valid Object Value(s)

This object will always return “time zero” as rows in the table are not dynamic in this implementation.

5.6.6 Traps

This section identifies Entity MIB Trap Definitions.

OIDs:

- entityMIBTrapPrefix - { entityMIBTraps 0 } 1.3.6.1.2.1.47.1.0

Table 65. ENTITY-MIB Traps

Object	OID	Trap Objects	Status	Supported
entConfigChange	{ entityMIBTrapPrefix 1 }	None	current	N

5.6.6.1 entConfigChange

An entConfigChange notification is generated when the value of entLastChangeTime changes. It can be utilized by an NMS to trigger logical/physical entity table maintenance polls.

An agent should not generate more than one entConfigChange “notification-event” in a given time interval (five seconds is the suggested default). A “notification-event” is the transmission of a single trap or inform PDU to a list of notification destinations.

If additional configuration changes occur within the throttling period, then notification-events for these changes should be suppressed by the agent until the current throttling period expires. At the end of a throttling period, one notification-event should be generated if any configuration changes occurred since the start of the throttling period. In such a case, another throttling period is started right away.

An NMS should periodically check the value of entLastChangeTime to detect any missed entConfigChange notification-events, for example, due to throttling or transmission loss.

5.7 ENTITY-SENSOR-MIB — RFC 3433

5.7.1 Introduction

This MIB describes managed objects for extending the Entity MIB (RFC 2737) (Section 5.6, Page 83)to provide generalized access to information related to physical sensors, which are often found in networking equipment (such as chassis temperature, fan RPM, power supply voltage). Physical sensors are represented in the Entity MIB with entPhysicalEntry (Section 5.6.5.1.1, Page 86) and an entPhysicalClass (Section 5.6.5.1.1.5, Page 101) value of "sensor(8)." The information provided in the ENTITY-SENSOR-MIB module (defined in this document) defines a sparse augmentation of the entPhysicalTable, for entries which represent physical sensors. The RFC used as a reference was dated December 16, 2002. The LAST-UPDATED for this MIB in the RFC was "200212160000Z."

5.7.2 Textual Conventions

The following textual conventions are made in this RFC:

- EntitySensorDataType - An object using this data type represents the Entity Sensor measurement data type associated with a physical sensor value. The actual data units are determined by examining an object of this type together with the associated EntitySensorDataScale object.

  An object of this type SHOULD be defined together with objects of type EntitySensorDataScale and EntitySensorPrecision. Together, associated objects of these three types are used to identify the semantics of an object of type EntitySensorValue. Valid values are:

  — SYNTAX INTEGER
  - other(1) - a measure other than those listed below
  - unknown(2) - unknown measurement, or arbitrary, relative numbers
  - voltsAC(3) - electric potential
  - voltsDC(4) - electric potential
  - amperes(5) - electric current
  - watts(6) - power
  - hertz(7) - frequency
  - celsius(8) - temperature
  - percentRH(9) - percent relative humidity
  - rpm(10) - shaft revolutions per minute
  - cmm(11) - cubic meters per minute (airflow)
  - truthvalue(12) - value takes: true(1) or false(2)

- EntitySensorDataScale - An object using this data type represents a data scaling factor, represented with an International System of Units (SI) prefix. The actual data units are determined by examining an object of this type together with the associated EntitySensorDataType object.

  An object of this type SHOULD be defined together with objects of type EntitySensorDataType and EntitySensorPrecision. Together, associated objects of these three types are used to identify the semantics of an object of type EntitySensorValue.

  — SYNTAX INTEGER
  - yocto(1) - 10^-24
  - zepto(2) - 10^-21
  - atto(3) - 10^-18
  - femto(4) - 10^-15
  - pico(5) - 10^-12
  - nano(6) - 10^-9

- micro(7) - 10^-6
- milli(8) - 10^-3
- units(9) - 10^0
- kilo(10) - 10^3
- mega(11) - 10^6
- giga(12) - 10^9
- tera(13) - 10^12
- exa(14) - 10^15
- peta(15) - 10^18
- zetta(16) - 10^21
- yotta(17) - 10^24

- EntitySensorPrecision - An object using this data type represents a sensor precision range.

  An object of this type SHOULD be defined together with objects of type EntitySensorDataType and EntitySensorDataScale. Together, associated objects of these three types are used to identify the semantics of an object of type EntitySensorValue.

  If an object of this type contains a value in the range 1 to 9, it represents the number of decimal places in the fractional part of an associated EntitySensorValue fixed- point number.

  If an object of this type contains a value in the range -8 to -1, it represents the number of accurate digits in the associated EntitySensorValue fixed-point number.

  The value zero indicates the associated EntitySensorValue object is not a fixed-point number.

  Agent implementors must choose a value for the associated EntitySensorPrecision object so that the precision and accuracy of the associated EntitySensorValue object is correctly indicated.

  For example, a physical entity representing a temperature sensor that can measure 0 degrees to 100 degrees C in 0.1 degree increments, +/- 0.05 degrees, would have an EntitySensorPrecision value of “1,” an EntitySensorDataScale value of units(9), and an EntitySensorValue ranging from “0” to “1000.” The EntitySensorValue would be interpreted as “degrees C * 10."

  — SYNTAX Integer32 (-8…9)

- EntitySensorValue - An object using this data type represents an Entity Sensor value. An object of this type SHOULD be defined together with objects of type EntitySensorDataType, EntitySensorDataScale and EntitySensorPrecision. Together, associated objects of those three types are used to identify the semantics of an object of this data type.

  The semantics of an object using this data type are determined by the value of the associated EntitySensorDataType object.

  If the associated EntitySensorDataType object is equal to voltsAC(3), voltsDC(4), amperes(5), watts(6), hertz(7), celsius(8), or cmm(11), then an object of this type MUST contain a fixed point number ranging from -999,999,999 to +999,999,999. The value -1000000000 indicates an underflow error. The value +1000000000 indicates an overflow error. The EntitySensorPrecision indicates how many fractional digits are represented in the associated EntitySensorValue object.

If the associated EntitySensorDataType object is equal to percentRH(9), then an object of this type MUST contain a number ranging from 0 to 100.

If the associated EntitySensorDataType object is equal to rpm(10), then an object of this type MUST contain a number ranging from -999,999,999 to +999,999,999.

If the associated EntitySensorDataType object is equal to truthvalue(12), then an object of this type MUST contain either the value true(1) or the value false(2). If the associated EntitySensorDataType object is equal to other(1) or unknown(2), then an object of this type MUST contain a number ranging from -1000000000 to 1000000000.

— SYNTAX Integer32 (-1000000000…1000000000)

- EntitySensorStatus - An object using this data type represents the operational status of a physical sensor.
  — SYNTAX INTEGER
  - ok(1) - Indicates that the agent can obtain the sensor value.
  - unavailable(2) - Indicates that the agent presently cannot obtain the sensor value.
  - nonoperational(3)- Indicates that the agent believes the sensor is broken. The sensor could have a hard failure (disconnected wire), or a soft failure such as out-of-range, jittery, or wildly fluctuating readings.

5.7.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- entitySensorMIB - { mib-2 99 } 1.3.6.1.2.1.99
- entitySensorObjects - { entitySensorMIB 1 } 1.3.6.1.2.1.99.1
- entitySensorNotifications - { entitySensorMIB 2 } 1.3.6.1.2.1.99.2

5.7.4 MIB Objects

Table 66. ENTITY-SENSOR_MIB Objects

Object	OID	Syntax	Access	Status	Supported
entPhySensorTable	{ entitySensorObjects 1 }	Sequence of entPhySensorEntry	not-accessible	current	Y

5.7.4.1 entPhySensorTable

This table contains one row per physical sensor represented by an associated row in the entPhysicalTable.

Entries are created in this table by the agent. An entry for each physical sensor SHOULD be created at the same time as the associated entPhysicalEntry. An entry SHOULD be destroyed if the associated entPhysicalEntry is destroyed.

Valid Object Value(s)

The following sensors are supported:

- Fan 1 Speed Sensor
- Fan 2 Speed Sensor
- Fan 3 Speed Sensor
- Fan 4 Speed Sensor[1]
- Intake Temperature Sensor
- Internal Temperature Sensor

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- entPhySensorTable - {entitySensorObjects 1 } 1.3.6.1.2.1.99.1.1
- entPhySensorEntry - {entPhySensorTable 1 } 1.3.6.1.2.1.99.1.1.1

Table 67. entPhySensorTable

Object	OID	Syntax	Access	Status	Supported
entPhySensorType	{ entPhySensorEntry 1 }	EntitySensorDataType	read-only	current	Y
entPhySensorScale	{ entPhySensorEntry 2 }	EntitySensorDataScale	read-only	current	Y
entPhySensorPrecision	{ entPhySensorEntry 3 }	EntitySensorPrecision	read-only	current	Y
entPhySensorValue	{ entPhySensorEntry 4 }	EntitySensorValue	read-only	current	Y
entPhySensorOperStatus	{ entPhySensorEntry 5 }	EntitySensorStatus	read-only	current	Y
entPhySensorUnitsDisplay	{ entPhySensorEntry 6 }	SnmpAdminString	read-only	current	Y
entPhySensorValueTimeStamp	{ entPhySensorEntry 7 }	TimeStamp	read-only	current	Y
entPhySensorValueUpdateRate	{ entPhySensorEntry 8 }	Unsigned32	read-only	current	Y

5.7.4.1.1 entPhySensorType

The type of data returned by the associated entPhySensorValue object.

This object SHOULD be set by the agent during entry creation, and the value SHOULD NOT change during operation.

5.7.4.1.2 entPhySensorScale

The exponent to apply to values returned by the associated entPhySensorValue object.

This object SHOULD be set by the agent during entry creation, and the value SHOULD NOT change during operation.

1. The A/C powered chassis has 4 fans; the D/C powered chassis has 3 fans.

5.7.4.1.3 entPhySensorPrecision

The number of decimal places of precision in fixed-point sensor values returned by the associated entPhySensorValue object.

This object SHOULD be set to “0” when the associated entPhySensorType value is not a fixed-point type: for example, percentRH(9), rpm(10), cmm(11), or truthvalue(12).

This object SHOULD be set by the agent during entry creation, and the value SHOULD NOT change during operation.

5.7.4.1.4 entPhySensorValue

The most recent measurement obtained by the agent for this sensor.

To correctly interpret the value of this object, the associated entPhySensorType, entPhySensorScale, and entPhySensorPrecision objects must also be examined.

5.7.4.1.5 entPhySensorOperStatus

The operational status of the sensor.

5.7.4.1.6 entPhySensorUnitsDisplay

A textual description of the data units that should be used in the display of entPhySensorValue.

5.7.4.1.7 entPhySensorValueTimeStamp

The value of sysUpTime at the time the status and/or value of this sensor was last obtained by the agent.

5.7.4.1.8 entPhySensorValueUpdateRate

An indication of the frequency that the agent updates the associated entPhySensorValue object, representing in milliseconds.

The value zero indicates:

- the sensor value is updated on demand (for example, when polled by the agent for a get-request)
- the sensor value is updated when the sensor value changes (event-driven)
- the agent does not know the update rate.

5.7.5 Conformance Information

5.7.5.1 Compliance

5.7.5.1.1 entitySensorCompliance

Describes the requirements for conformance to the Entity Sensor MIB module.

The following are mandatory groups:

- MODULE this module
  - entitySensorValueGroup
- MODULE ENTITY-MIB
  - entityPhysicalGroup

5.7.5.2 Conformance

Table 68. ENITITY-SENSOR-MIB Conformance

Group	Description	Objects	Supported
entitySensorValueGroup	A collection of objects representing physical entity sensor information.	entPhySensorType	Y
		entPhySensorScale	Y
		entPhySensorPrecision	Y
		entPhySensorValue	Y
		entPhySensorOperStatus	Y
		entPhySensorUnitsDisplay	Y
		entPhySensorValueTimeStamp	Y
		entPhySensorValueUpdateRate	Y

5.8 EtherLike-MIB — RFC 2665

5.8.1 Introduction

This MIB describes objects for managing Ethernet-like interfaces. The RFC used as a reference was dated August 1999. The LAST-UPDATED for the Ethernet-Like-MIB was “9908240400Z.”

5.8.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- transmission - { mib-2 10 } 1.3.6.1.2.1.10
- etherMIB - { mib-2 35 } 1.3.6.1.2.1.35
- etherMIBObjects - { etherMIB 1 } 1.3.6.1.2.1.35.1
- dot3 - { transmission 7 } 1.3.6.1.2.1.10.7
- dot3StatsTable - { dot3 2 } 1.3.6.1.2.1.10.7.2
- dot3CollTable - { dot3 5 } 1.3.6.1.2.1.10.7.5
- dot3Tests - { dot3 6 } 1.3.6.1.2.1.10.7.6
- dot3Errors - { dot3 7 } 1.3.6.1.2.1.10.7.7
- dot3ControlTable - { dot3 9 } 1.3.6.1.2.1.10.7.9
- dot3PauseTable - { dot3 10 } 1.3.6.1.2.1.10.7.10

5.8.3 MIB Objects

The following groups supported in this MIB are as follow:

- Ethernet-like Statistics Group
- Ethernet-like Collision Statistics Group
- Ethernet-like Control Group
- Ethernet-like Pause Group
- Ethernet-like Test Group
- Ethernet-like Errors Group

The table below identifies the objects called out in this MIB. Any table object that has elements that are supported are indicated by a “Y” in the Supported column.

Table 69. EtherLike-MIB Objects (1 of 2)

Object	OID	Syntax	Access	Status	Supported
dot3StatsTable	{ dot3 2 }	Sequence of dot3StatsEntry	not-accessible	current	Y
dot3CollTable	{ dot3 5 }	Sequence of dot3CollEntry	not-accessible	current	N
dot3ControlTable	{ dot3 9 }	Sequence of dot3ControlEntry	not-accessible	current	N
dot3PauseTable	{ dot3 10 }	Sequence of dot3PauseEntry	not-accessible	current	Y
dot3Tests	{ dot3 6 }	N/A[a]	not-accessible	current	N

Table 69. EtherLike-MIB Objects (2 of 2)

Object	OID	Syntax	Access	Status	Supported
dot3Errors	{ dot3 7 }	N/A[a]	not-accessible	current	N

a. These are simply object grouped under the respective OID.

5.8.3.1 Ethernet-like Statistics Group

This group defines statistics for collection of Ethernet-like interfaces attached to a particular system. The statistics are arranged in a table object known as dot3StatsTable.

5.8.3.1.1 dot3StatsTable

The dot3Stats table contains Statistics for a collection of Ethernet-like interfaces attached to a particular system. There is one row in this table for each Ethernet-like interface in the system. This table is a sequence of dot3StatsEntry object type.

The dot2StatsEntry object is an entry containing statistics for a particular interface to an Ethernet-like medium.

Index:

- dot3StatsIndex

OIDs:

- dot3StatsTable - { dot3 2 } 1.3.6.1.2.1.10.7.2
- dot3StatsEntry - { dot3StatsTable 1 } 1.3.6.1.2.1.10.7.2.1

Table 70. dot3StatsTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
dot3StatsIndex	{ dot3StatsEntry 1 }	InterfaceIndex	read-only	current	Y
dot3StatsAlignmentErrors	{ dot3StatsEntry 2 }	Counter32	read-only	current	Y
dot3StatsFCSErrors	{ dot3StatsEntry 3 }	Counter32	read-only	current	Y
dot3StatsSingleCollisionFrames	{ dot3StatsEntry 4 }	Counter32	read-only	current	Y
dot3StatsMultipleCollisionsFrames	{ dot3StatsEntry 5 }	Counter32	read-only	current	Y
dot3StatsSQETestErrors	{ dot3StatsEntry 6 }	Counter32	read-only	current	N
dot3StatsDeferredTransmissions	{ dot3StatsEntry 7 }	Counter32	read-only	current	Y
dot3StatsLateCollisions	{ dot3StatsEntry 8 }	Counter32	read-only	current	Y
dot3StatsExcessiveCollisions	{ dot3StatsEntry 9 }	Counter32	read-only	current	Y
dot3StatsInternalMacTransmitErrors	{ dot3StatsEntry 10 }	Counter32	read-only	current	N
dot3StatsCarrierSenseErrors	{ dot3StatsEntry 11 }	Counter32	read-only	current	N
	{ dot3StatsEntry 12 }[a]				
dot3StatsFrameTooLongs	{ dot3StatsEntry 13 }	Counter32	read-only	current	Y
	{ dot3StatsEntry 14 }[a]				
	{ dot3StatsEntry 15 }[a]				

Table 70. dot3StatsTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
dot3StatsInternalMacReceiveErrors	{ dot3StatsEntry 16 }	Counter32	read-only	current	N
dot3StatsEtherChipSet	{ dot3StatsEntry 17 }	OBJECT IDENTIFIER	read-only	deprecated	N
dot3StatsSymbolErrors	{ dot3StatsEntry 18 }	Counter32	read-only	current	N
dot3StatsDuplexStatus	{ dot3StatsEntry 19 }	INTEGER unknown(1) halfDuplex(2) fullDeplex(3)	read-only	current	Y

a. This OID is not used.

5.8.3.1.1.1 dot3StatsIndex

An index value that uniquely identifies an interface to an Ethernet-like medium. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex.

5.8.3.1.1.2 dot3StatsAlignmentErrors

A count of frames received on a particular interface that are not an integral number of octets in length and do not pass the FCS check.

The count represented by an instance of this object is incremented when the alignmentError status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions obtain are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.

This counter does not increment for 8-bit wide group encoding schemes. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.3 dot3StatsFCSErrors

A count of frames received on a particular interface that are an integral number of octets in length but do not pass the FCS check. This count does not include frames received with frame-too-long or frame-too-short error.

The count represented by an instance of this object is incremented when the frameCheckError status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions obtain are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.

Note: Coding errors detected by the physical layer for speeds above 10 Mbps will cause the frame to fail the FCS check. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.4 dot3StatsSingleCollisionFrames

A count of successfully transmitted frames on a particular interface for which transmission is inhibited by exactly one collision.

A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts, ifOutMulticastPkts, or ifOutBroadcastPkts, and is not counted by the corresponding instance of the dot3StatsMultipleCollisionFrames object.

This counter does not increment when the interface is operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.5 dot3StatsMultipleCollisionFrames

A count of successfully transmitted frames on a particular interface for which transmission is inhibited by more than one collision.

A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts, ifOutMulticastPkts, or ifOutBroadcastPkts, and is not counted by the corresponding instance of the dot3StatsSingleCollisionFrames object.

This counter does not increment when the interface is operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.6 dot3StatsSQETestErrors

This object is not supported.

A count of times that the SQE TEST ERROR message is generated by the PLS sublayer for a particular interface. The SQE TEST ERROR is set in accordance with the rules for verification of the SQE detection mechanism in the PLS Carrier Sense Function as described in IEEE Std. 802.3, 1998 Edition, section 7.2.4.6.

This counter does not increment on interfaces operating at speeds greater than 10 Mbps, or on interfaces operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.7 dot3StatsDeferredTransmissions

A count of frames for which the first transmission attempt on a particular interface is delayed because the medium is busy. The count represented by an instance of this object does not include frames involved in collisions.

This counter does not increment when the interface is operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.8 dot3StatsLateCollisions

The number of times that a collision is detected on a particular interface later than one slotTime into the transmission of a packet.

A (late) collision included in a count represented by an instance of this object is also considered as a (generic) collision for purposes of other collision-related statistics.

This counter does not increment when the interface is operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.9 dot3StatsExcessiveCollisions

A count of frames for which transmission on a particular interface fails due to excessive collisions.

This counter does not increment when the interface is operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.10 dot3StatsInternalMacTransmitErrors

This object is not supported.

A count of frames for which transmission on a particular interface fails due to an internal MAC sublayer transmit error. A frame is only counted by an instance of this object if it is not counted by the corresponding instance of either the dot3StatsLateCollisions object, the dot3StatsExcessiveCollisions object, or the dot3StatsCarrierSenseErrors object.

The precise meaning of the count represented by an instance of this object is implementation- specific. In particular, an instance of this object may represent a count of transmission errors on a particular interface that are not otherwise counted.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.11 dot3StatsCarrierSenseErrors

This object is not supported.

The number of times that the carrier sense condition was lost or never asserted when attempting to transmit a frame on a particular interface.

The count represented by an instance of this object is incremented at most once per transmission attempt, even if the carrier sense condition fluctuates during a transmission attempt.

This counter does not increment when the interface is operating in full-duplex mode.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.12 dot3StatsFrameTooLongs

A count of frames received on a particular interface that exceed the maximum permitted frame size.

The count represented by an instance of this object is incremented when the frameTooLong status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions obtain are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.13 dot3StatsInternalMacReceiveErrors

This object is not supported.

A count of frames for which reception on a particular interface fails due to an internal MAC sublayer receive error. A frame is only counted by an instance of this object if it is not counted by the corresponding instance of either the dot3StatsFrameTooLongs object, the dot3StatsAlignmentErrors object, or the dot3StatsFCSErrors object.

The precise meaning of the count represented by an instance of this object is implementation specific. In particular, an instance of this object may represent a count of receive errors on a particular interface that are not otherwise counted.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.14 dot3StatsEtherChipSet

THIS OBJECT IS DEPRECATED

This object is not supported.

This object contains an OBJECT IDENTIFIER which identifies the chipset used to realize the interface. Ethernet-like interfaces are typically built out of several different chips. The MIB implementor is presented with a decision of which chip to identify via this object. The implementor should identify the chip which is usually called the Medium Access Control chip. If no such chip is easily identifiable, the implementor should identify the chip which actually gathers the transmit and receive statistics and error indications. This would allow a manager station to correlate the statistics and the chip generating them, giving it the ability to take into account any known anomalies in the chip.

5.8.3.1.1.15 dot3StatsSymbolErrors

This object is not supported.

For an interface operating at 100 Mbps, the number of times there was an invalid data symbol when a valid carrier was present.

For an interface operating in half-duplex mode at 1000 Mbps, the number of times the receiving media is non-idle (a carrier event) for a period of time equal to or greater than slotTime, and during which there was at least one occurrence of an event that causes the PHY to indicate “Data reception error” or “carrier extend error” on the GMII.

For an interface operating in full-duplex mode at 1000 Mbps, the number of times the receiving media is non-idle a carrier event) for a period of time equal to or greater than minFrameSize, and during which there was at least one occurrence of an event that causes the PHY to indicate “Data reception error” on the GMII.

The count represented by an instance of this object is incremented at most once per carrier event, even if multiple symbol errors occur during the carrier event. This count does not increment if a collision is present.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.1.1.16 dot3StatsDuplexStatus

The current mode of operation of the MAC entity. “unknown” indicates that the current duplex mode could not be determined.

Management control of the duplex mode is accomplished through the MAU MIB. When an interface does not support autonegotiation, or when autonegotiation is not enabled, the duplex mode is controlled using ifMauDefaultType. When autonegotiation is supported and enabled, duplex mode is controlled using ifMauAutoNegAdvertisedBits. In either case, the currently operating duplex mode is reflected both in this object and in ifMauType.

Note that this object provides redundant information with ifMauType. Normally, redundant objects are discouraged. However, in this instance, it allows a management application to determine the duplex status of an interface without having to know every possible value of ifMauType. This was felt to be sufficiently valuable to justify the redundancy.

5.8.3.2 Ethernet-like Collision Statistics Group

This group, which consists of the dot3CollTable, is optional and is only appropriate for all systems which have the necessary metering.

This group is not supported. The RFC states that it is no loner in use.

OIDs:

- dot3CollTable - { dot3 5 } 1.3.6.1.2.1.10.7.5
- dot3CollEntry - { dot3CollTable 1 } 1.3.6.1.2.1.10.7.5.1

Table 71. dot3CollTable

Object	OID	Syntax	Access	Status	Supported
	{ dot3CollEntry 1 }[a]				
dot3CollCount	{ dot3CollEntry 2 }	INTEGER(1…16)	not-accessible	current	N
dot3CollFrequencies	{ dot3CollEntry 3 }	Counter32	read-only	current	N

a. This OID is no longer in use.

5.8.3.3 Ethernet-like Control Group

This group defines objects for 802.3 control functions.

5.8.3.3.1 dot3ControlTable

These functions are arranged in a table object known as dot3ControlTable. There is one row in this table for each Ethernet-like interface in the system. This table is a sequence of dot1ControlEntry type object. The index for this table is the dot3StatsIndex described in Section 5.8.3.1.1.1, Page 127.

The dot3ControlEntry object is an entry in the table containing information about the MAC Control sublayer on a single Ethernet-like interface.

Index:

- dot3StatsIndex (Section 5.8.3.1.1.1, Page 127)

OIDs:

- dot3ControlTable - { dot3 9 } 1.3.6.1.2.1.10.7.9
- dot3ControlEntry - { dot3ControlTable 1 } 1.3.6.1.2.1.10.7.9.1

,

Table 72. dot3ControlTable

Object	OID	Syntax	Access	Status	Supported
dot3ControlFunctionsSupported	{ dot3ControlEntry 1 }	BITS pause(0)	read-only	current	N
dot3ControlInUnknownOpcodes	{ dot3ControlEntry 2 }	Counter32	read-only	current	N

5.8.3.3.1.1 dot3ControlFunctions

A list of the possible MAC Control functions implemented for this interface.

Pause is supported so the pause bit will be set.

5.8.3.3.1.2 dot3ControlInUnknownOpcodes

A count of MAC Control frames received on this interface that contain an opcode that is not supported by this device.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.4 Ethernet-like Pause Group

This group defines objects containing information about MAC Control PAUSE functions. The information is arranged in a table object known as dot3PauseTable.

5.8.3.4.1 dot3PauseTable

The dot3PauseTable table of descriptive and status information about the MAC Control PAUSE function on the net-like interfaces attached to a particular system. There is one row in this table for each Ethernet-like interface in the system which supports the MAC Control PAUSE function (that is, the “pause” bit in the corresponding instance of dot3ControlFunctionsSupported is set). If some, but not all, of the Ethernet-like interfaces in the system implement the MAC Control PAUSE function (for example, if some interfaces only support half-duplex), there will be fewer rows in this table than in the dot3StatsTable. This table is a sequence of dot3PauseEntry type object.

The dot3PauseEntry object defines an entry in the table, containing information about the MAC Control PAUSE function on a single Ethernet-like interface.

Index:

- dot3StatsIndex (Section 5.8.3.1.1.1, Page 127)

OIDs:

- dot3PauseTable - { dot3 10 } 1.3.6.1.2.1.10.7.10
- dot3PauseEntry - { dot3PauseTable 1 } 1.3.6.1.2.1.10.7.10.1

,

Table 73. dot3PauseTable

Object	OID	Syntax	Access	Status	Supported
dot3PauseAdminMode	{ dot3PauseEntry 1 }	INTEGER disabled(1) enabledXmit(2) enabledRcv(3) enabledXmitAndRcv(4)	read-write	current	Y
dot3PauseOperMode	{ dot3PauseEntry 2 }	INTEGER disabled(1) enabledXmit(2) enabledRcv(3) enabledXmitAndRcv(4)	read-only	current	Y
dot3InPauseFrames	{ dot3PauseEntry 3 }	Counter32	read-only	current	N
dot3OutPauseFrames	{ dot3PauseEntry 4 }	Counter32	read-only	current	N

5.8.3.4.1.1 dot3PauseAdminMode

This object is used to configure the default administrative PAUSE mode for this interface.

This object represents the administratively-configured PAUSE mode for this interface. If auto-negotiation is not enabled or is not implemented for the active MAU attached to this interface, the value of this object determines the operational PAUSE mode of the interface whenever it is operating in full-duplex mode. In this case, a set to this object will force the interface into the specified mode.

If auto-negotiation is implemented and enabled for the MAU attached to this interface, the PAUSE mode for this interface is determined by auto-negotiation, and the value of this object denotes the mode to which the interface will automatically revert if/when auto-negotiation is later disabled. Note that when auto-negotiation is running, administrative control of the PAUSE mode may be accomplished using the ifMauAutoNegCapAdvertisedBits object in the MAU-MIB.

Note that the value of this object is ignored when the interface is not operating in full-duplex mode.

Valid Object Value(s)

- disabled(1)
- enableXmitAndRcv(4)

An attempt to set this object to “enabledXmit(2)” or “enabledRcv(3)” will fail on interfaces that do not support operation at greater than 100 Mbps.

5.8.3.4.1.2 dot3PauseOperMode

This object reflects the PAUSE mode currently in use on this interface, as determined by either (1) the result of the auto-negotiation function or (2) if auto-negotiation is not enabled or is not implemented for the active MAU attached to this interface, by the value of dot3PauseAdminMode. Interfaces operating at 100 Mbps or less will never return “enabledXmit(2)” or “enabledRcv(3).” Interfaces operating in half-duplex mode will always return “disabled(1).” Interfaces on which auto-negotiation is enabled but not yet completed should return the value “disabled(1).”

Valid Object Value(s)

- disabled(1)
- enabledXmitAndRcv(4)

5.8.3.4.1.3 dot3InPauseFrames

A count of MAC Control frames received on this interface with an opcode indicating the PAUSE operation.

This counter does not increment when the interface is operating in half-duplex mode. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.4.1.4 dot3OutPauseFrames

A count of MAC Control frames transmitted on this interface with an opcode indicating the PAUSE operation.

This counter does not increment when the interface is operating in half-duplex mode. Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.8.3.5 Ethernet-like Test Group

This group will not be supported. No Time-Domain Reflectometry (TDR) tests or Loopback tests are supported for Ethernet in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

OIDs:

- dot3Tests - { dot3 6 } 1.3.6.1.2.1.10.7.6

,

Table 74. dot3Tests

Object	OID	Syntax	Access	Status	Supported
dot3TestTdr	{ dot3Tests 1 }	unspecified	unspecified	current	N
dot3TestLoopBack	{ dot3Tests 2 }	unspecified	unspecified	current	N

5.8.3.6 Ethernet-like Errors Group

This group will not be supported due to lack of support for the Ethernet-like Test Group.

OIDs:

- dot3Errors - { dot3 7 } 1.3.6.1.2.1.10.7.7

,

Table 75. dot3Tests

Object	OID	Syntax	Access	Status	Supported
dot3ErrInitError	{ dot3Errors 1 }	unspecified	unspecified	current	N
dot3ErrorLoopbackError	{ dot3Errors 2 }	unspecified	unspecified	current	N

5.9 MAU-MIB — RFC 2668

5.9.1 Introduction

This MIB describes objects for managing IEEE 802.3 Medium Attachment Units (MAUs). The RFC used as a reference was dated August 1999. The LAST-UPDATED for the MAU-MIB was “9908240400Z.”

5.9.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- transmission - { mib-2 10 } 1.2.6.1.2.1.10
- snmpDot3MauMgt - { mib-2 26 } 1.2.6.1.2.1.26
- snmpDot3MauTraps - { snmpDot3MauMgt 0 } 1.2.6.1.2.1.26.0
- dot3RpMauBasicGroup - { snmpDot3MauMgt 1 } 1.2.6.1.2.1.26.1
- dot3IfMauBasicGroup - { snmpDot3MauMgt 2 } 1.2.6.1.2.1.26.2
- dot3BroadMauBasicGroup - { snmpDot3MauMgt 3 } 1.2.6.1.2.1.26.3
- dot3MauType - { snmpDot3MauMgt 4 } 1.2.6.1.2.1.26.4
- dot3IfMauAutoNegGroup - { snmpDot3MauMgt 5 } 1.2.6.1.2.1.26.5
- mauMod - { snmpDot3MauMgt 6 } 1.2.6.1.2.1.26.6

5.9.3 Textual Conventions

- JackType - Common enumeration values for repeater and interface MAU jack types.
  - SYNTAX INTEGER
    - other(1)
    - rj45(2),
    - rj45S(3) - rj45 shielded
    - db9(4)
    - bnc(5)
    - fAUI(6) - female aui
    - mAUI(7) - male aui
    - fiberSC(8)
    - fiberMIC(9)
    - fiberST(10)
    - telco(11),
    - mtrj(12) - fiber MT-RJ
    - hssdc(13) - fiber channel style-2

The following are object identities for MAU types. See rpMauType and ifMauType for usage.

- dot3MauTypeAUI - { dot3MauType 1 } - no internal MAU, view from AUI
- dot3MauType10Base5 - { dot3MauType 2 } - thick coax MAU (per 802.3 section 8)
- dot3MauTypeFoirl - { dot3MauType 3 } - FOIRL MAU (per 802.3 section 9.9)
- dot3MauType10Base2 - { dot3MauType 4 } - thin coax MAU (per 802.3 section 10)
- dot3MauType10BaseT - { dot3MauType 5 } - UTP MAU (per 802.3 section 14)
  Note that it is strongly recommended that agents return either dot3MauType10BaseTHD

or dot3MauType10BaseTFD if the duplex mode is known. However, management applications should e prepared to receive this MAU type value from older agent implementations.

- dot3MauType10BaseFP - { dot3MauType 6 } - passive fiber MAU (per 802.3 section 16)
- dot3MauType10BaseFB - { dot3MauType 7 } - sync fiber MAU (per 802.3 section 17)
- dot3MauType10BaseFL - { dot3MauType 8 } - async fiber MAU (per 802.3 section 18) Note that it is strongly recommended that agents return either dot3MauType10BaseFLHD or dot3MauType10BaseFLFD if the duplex mode is known. However, management applications should be prepared to receive this MAU type value from older agent implementations.
- dot3MauType10Broad36 - { dot3MauType 9 } - broadband DTE MAU (per 802.3 section 11) Note that 10BROAD36 MAUs can be attached to interfaces but not to repeaters.
- dot3MauType10BaseTHD - { dot3MauType 10 } - UTP MAU (per 802.3 section 14), half duplex mode
- dot3MauType10BaseTFD - { dot3MauType 11 } - UTP MAU (per 802.3 section 14), full duplex mode
- dot3MauType10BaseFLHD - { dot3MauType 12 } - async fiber MAU (per 802.3 section 18), half duplex mode
- dot3MauType10BaseFLFD - { dot3MauType 13 } - async fiber MAU (per 802.3 section 18), full duplex mode
- dot3MauType100BaseT4 - { dot3MauType 14 } - 4 pair categ. 3 UTP (per 802.3 section 23)
- dot3MauType100BaseTXHD - { dot3MauType 15 } - 2 pair categ. 5 UTP (per 802.3 section 25), half duplex mode
- dot3MauType100BaseTXFD - { dot3MauType 16 } - 2 pair categ. 5 UTP (per 802.3 section 25),full duplex mode
- dot3MauType100BaseFXHD - { dot3MauType 17 } - X fiber over PMT (per 802.3 section 26), half duplex mode
- dot3MauType100BaseFXFD - { dot3MauType 18 } - X fiber over PMT (per 802.3 section 26), full duplex mode
- dot3MauType100BaseT2HD - { dot3MauType 19 } - 2 pair categ. 3 UTP (per 802.3 section 32), half duplex mode
- dot3MauType100BaseT2FD - { dot3MauType 20 } - 2 pair categ. 3 UTP (per 802.3 section 32), full duplex mode
- dot3MauType1000BaseXHD - { dot3MauType 21 } - PCS/PMA (per 802.3 section 36), unknown PMD, half duplex mode
- dot3MauType1000BaseXFD - { dot3MauType 22 } - PCS/PMA (per 802.3 section 36), unknown PMD, full duplex mode
- dot3MauType1000BaseLXHD - { dot3MauType 23 } - Fiber over long-wavelength laser (per 802.3 section 38), half duplex mode
- dot3MauType1000BaseLXFD - { dot3MauType 24 } - Fiber over long-wavelength laser (per 802.3 section 38), full duplex mode
- dot3MauType1000BaseSXHD - { dot3MauType 25 } - Fiber over short-wavelength laser (per 802.3 section 38), half duplex mode

- dot3MauType1000BaseSXFD - { dot3MauType 26 } - Fiber over short-wavelength laser (per 802.3 section 38), full duplex mode
- dot3MauType1000BaseCXHD - { dot3MauType 27 } - Copper over 150-Ohm balanced cable (per 802.3 section 39), half duplex mode
- dot3MauType1000BaseCXFD - { dot3MauType 28 } - Copper over 150-Ohm balanced cable (per 802.3 section 39), full duplex mode
- dot3MauType1000BaseTHD - { dot3MauType 29 } - Four-pair Category 5 UTP (per 802.3 section 40), half duplex mode
- dot3MauType1000BaseTFD - { dot3MauType 30 } - Four-pair Category 5 UTP (per 802.3 section 40), full duplex mode

5.9.4 MIB Objects

The table below identifies the objects called out in this MIB. Any table object that has elements that are supported are indicated by a “Y” in the Supported column.

Table 76. MAU-MIB Objects

Object	OID	Syntax	Access	Status	Supported
rpMauTable	{ dot3RpMauBasicGroup 1 }	Sequence of rpMauEntry	not-accessible	current	N
rpJackTable	{ dot3RpMauBasicGroup 2 }	Sequence of rpJackEntry	not-accessible	current	N
ifMauTable	{ dot3IfMauBasicGrup 1 }	Sequence of ifMauEntry	not-accessible	current	Y
ifJackTable	{ dot3IfMauBasicGrup 2 }	Sequence of ifJackEntry	not-accessible	current	Y
ifMauAutoNegTable	{ dot3IfMauAutoNegGroup 1 }	Sequence of ifMauAutoNegEntry	not-accessible	current	Y
broadMauBasicTable	{ dot3BroadMauBasicGroup 1 }	Sequence of broadMauBasicEntry	not-accessible	deprecated	N

5.9.4.1 dot3RpMauBasicGroup

No objects in this group are supported since the BitStorm 2600 and GranDSLAM 4200 IP DSLAM do not act like a repeater.

5.9.4.1.1 rpMauTable

Table of descriptive and status information about the MAU(s) attached to the ports of a repeater. The table is a sequence of rpMauEntry where an entry contains information about a single MAU.

This table will not be supported since the BitStorm 2600 and GranDSLAM 4200 IP DSLAM do not act like a repeater.

Indices:

- rpMauGroupIndex
- rpMauPortIndex
- rpMauIndex

OIDs:

- rpMauTable - { dot3RpMauBasicGroup 1 } 1.2.6.1.2.1.26.1.1
- rpMauEntry - { rpMauTable 1 } 1.2.6.1.2.1.26.1.1.1

Table 77. rpMauTable

Object	OID	Syntax	Access	Status	Supported
rpMauGroupIndex	{ rpMauEntry 1 }	Integer32 (1…2147483647)	read-only	current	N
rpMauPortIndex	{ rpMauEntry 2 }	Integer32 (1…2147483647)	read-only	current	N
rpMauIndex	{ rpMauEntry 3 }	Integer32(1…2147483647)	read-only	current	N
rpMauType	{ rpMauEntry 4 }	OBJECT IDENTIFIER	read-only	current	N
rpMauStatus	{ rpMauEntry 5 }	INTEGER other(1) unknown(2) operational(3) standby(4) shutdown(5) reset(6)	read-write	current	N
rpMauMediaAvailable	{ rpMauEntry 6 }	INTEGER other(1) unknown(2) available(3) notAvailable(4) remoteFault(5) invalidSignal(6) remoteJabber(7) remoteLinkLoss(8) remoteTest(9) offline(10) autoNegError(11)	read-only	current	N
rpMauMediaAvailableStateExits	{ rpMauEntry 7 }	Counter32	read-only	current	N
rpMauJabberState	{ rpMauEntry 8 }	INTEGER other(1) unknown(2) noJabber(3) jabbering(4)	read-only	current	N
rpMauJabberingStateEnters	{ rpMauEntry 9 }	Counter32	read-only	current	N
rpMauFalseCarriers	{ rpMauEntry 10 }	Counter32	read-only	current	N

5.9.4.1.1.1 rpMauGroupIndex

This variable uniquely identifies the group containing the port to which the MAU described by this entry is connected.

Note: In practice, a group will generally be a field-replaceable unit (such as a module, card, or board) that can fit in the physical system enclosure, and the group number will correspond to a number marked on the physical enclosure.

The group denoted by a particular value of this object is the same as the group denoted by the same value of rptrGroupIndex.

5.9.4.1.1.2 rpMauPortIndex

This variable uniquely identifies the repeater port within group rpMauGroupIndex to which the MAU described by this entry is connected.

5.9.4.1.1.3 rpMauIndex

This variable uniquely identifies the MAU described by this entry from among other MAUs connected to the same port (rpMauPortIndex).

5.9.4.1.1.4 rpMauType

This object identifies the MAU type. An initial set of MAU types are defined above. The assignment of OBJECT IDENTIFIERs to new types of MAUs is managed by the IANA. If the MAU type is unknown, the object identifier unknownMauType OBJECT IDENTIFIER = { 0 0 } is returned. Note that unknownMauType is a syntactically valid object identifier, and any conformant implementation of ASN.1 and the BER must be able to generate and recognize this value.

5.9.4.1.1.5 rpMauStatus

The current state of the MAU. This object MAY be implemented as a read-only object by those agents and MAUs that do not implement software control of the MAU state. Some agents may not support setting the value of this object to some of the enumerated values.

- other(1) - The MAU is in a state other than one of the states 2 through 6 below
- unknown(2) - The MAU's true state is unknown; for example, when it is being initialized
- operational(3) - The MAU is fully functional, operates, and passes signals to its attached DTE or repeater port in accordance to its specification
- standby(4) - This state forces DI and CI to idle and the media transmitter to idle or fault, if supported.
- shutdown(5) - This state assumes the same condition on DI, CI, and the media transmitter as though it were powered down or not connected. The MAU MAY return other(1) value for the rpMauJabberState and rpMauMediaAvailable objects when it is in this state. For an AUI, this state will remove power from the AUI
- reset(6) - This state resets the MAU in the same manner as a power-off, power-on cycle of at least one-half second would. The agent is not required to return the value reset (6).

Setting this variable to the value operational(3), standby(4), or shutdown(5) causes the MAU to assume the respective state except that setting a mixing-type MAU or an AUI to standby(4) will cause the MAU to enter the shutdown state.

5.9.4.1.1.6 rpMauMediaAvailable

If the MAU is a link or fiber type (FOIRL, 10BASE-T, 10BASE-F) then this is equivalent to the link test fail state/low light function. For an AUI or a coax (including broadband) MAU this indicates whether or not loopback is detected on the DI circuit. The value of this attribute persists between packets for MAU types AUI, 10BASE5, 10BASE2, 10BROAD36, and 10BASE-FP.

- other(1) - Returned if the mediaAvailable state is not one of 2 through 11
- unknown(2) - Returned when the MAU's true state is unknown; for example, when it is being initialized. At power-up or following a reset, the value of this attribute is unknown for AUI, coax, and 10BASE-FP MAUs. For these MAUs loopback is tested on each transmission during which no collision is detected. If DI is receiving input when DO returns to IDL after a transmission and there has been no collision during the transmission, then loopback will be detected. The value of this attribute will only change during non-collided transmissions for AUI, coax, and 10BASE-FP MAUs.
- available(3) - Indicates that the link, light, or loopback is normal.
- notAvailable(4) - Indicates link loss, low light, or no loopback.
- remoteFault(5) - Indicates that a fault has been detected at the remote end of the link. This value applies to 10BASE-FB, 100BASE-T4 Far End Fault Indication and non-specified remote faults from a system running auto-negotiation.

- invalidSignal(6) - Indicates that an invalid signal has been received from the other end of the link. InvalidSignal(6) applies only to MAUs of type 10BASE-FB.
- remoteJabber(7) - See description below.
- remoteLinkLoss(8) - See description below.
- remoteTest(9) - See description below.
- offline(10) - See description below.
- autoNegError(11) - See description below.

For 100Mbps and 1000Mbps MAUs, the enumerations match the states within the respective link integrity state diagrams, fig 32-16, 23-12 and 24-15 of sections 32, 23 and 24 of the IEEE Std 803.2, 1998 Edition specification. Any MAU which implements management of auto-negotiation will map remote fault indication to remote fault.

Where an IEEE Std 802.3u-1995 clause 22 MII is present, a logic one in the remote fault bit (reference section 22.2.4.2.8 of that document) maps to the value remoteFault(5), and a logic zero in the link status bit (reference section 22.2.4.2.10 of that document) maps to the value notAvailable(4). The value notAvailable(4) takes precedence over the value remoteFault(5).

Any MAU that implements management of clause 37 Auto-Negotiation will map the received Remote Fault (RF1 and RF2) bit values for Offline to offline(10), Link Failure to remoteFault(5) and Auto-Negotiation Error to autoNegError(11).

5.9.4.1.1.7 rpMauMediaAvailableStateExits

A count of the number of times that rpMauMediaAvailable for this MAU instance leaves the state available(3).

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of rptrMonitorPortLastChange.

5.9.4.1.1.8 rpMauJabberState

- other(1) - The value other(1) is returned if the jabber state is not 2, 3, or 4. The agent MUST always return other(1) for MAU type dot3MauTypeAUI.
- unknown(2) - The value unknown(2) is returned when the MAU's true state is unknown; for example, when it is being initialized.
- noJabber(3) - If the MAU is not jabbering the agent returns noJabber(3). This is the “normal” state.
- jabbering(4) - If the MAU is in jabber state the agent returns the jabbering(4) value.

5.9.4.1.1.9 rpMauJabberingStateEnters

A count of the number of times that mauJabberState for this MAU instance enters the state jabbering(4). For MAUs of type dot3MauTypeAUI, dot3MauType100BaseT4, dot3MauType100BaseTX, dot3MauType100BaseFX and all 1000Mbps types, this counter will always indicate zero.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of rptrMonitorPortLastChange.

5.9.4.1.1.10 rpMauFalseCarriers

A count of the number of false carrier events during IDLE in 100BASE-X links. This counter does not increment at the symbol rate. It can increment after a valid carrier completion at a maximum rate of once per 100 ms until the next carrier event.

This counter increments only for MAUs of type dot3MauType100BaseT4, dot3MauType100BaseTX, and dot3MauType100BaseFX and all 1000Mbps types. For all other MAU types, this counter will always indicate zero.

The approximate minimum time for rollover of this counter is 7.4 hours.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of rptrMonitorPortLastChange.

5.9.4.1.2 rpJackTable

The rpJackTable applies to MAUs attached to repeaters which have one or more external jacks (connectors).

Information about the external jacks attached to MAUs attached to the ports of a repeater. This table is a sequence of rpJackEntry where an entry contains information about a particular jack.

This table will not be supported since the BitStorm 2600 and GranDSLAM 4200 IP DSLAM do not act like a repeater.

Indices:

- rpMauGroupIndex (Section 5.9.4.1.1.1, Page 138)
- rpMauPortIndex (Section 5.9.4.1.1.2, Page 139)
- rpMauType (Section 5.9.4.1.1.4, Page 139)
- rpJackIndex

OIDs:

- rpJackTable - { dot3RpMauBasicGroup 2 } 1.2.6.1.2.1.26.1.2
- rpJackEntry - { rpJackTable 1 } 1.2.6.1.2.1.26.1.2.1

Table 78. rpJackTable

Object	OID	Syntax	Access	Status	Supported
rpJackIndex	{ rpJackEntry 1 }	Integer32 (1…2147483647)	read-only	current	N
rpJackType	{ rpJackEntry 2 }	JackType	read-only	current	N

5.9.4.1.2.1 rpJackIndex

This variable uniquely identifies the jack described by this entry from among other jacks attached to the same MAU (rpMauIndex).

5.9.4.1.2.2 rpJackType

The jack connector type, as it appears on the outside of the system.

5.9.4.2 dot3IfMauBasicGroup

5.9.4.2.1 ifMauTable

Table of descriptive and status information about MAU(s) attached to an interface.

This table is a sequence of IfMauEntry which consists of information about a single MAU.

Indices:

- ifMauIfIndex
- ifMauIndex

OIDs:

- ifMauTable - {dot3IfMauBasicGroup 1 } 1.2.6.1.2.1.26.2.1
- ifMauEntry - {ifMauTable 1 } 1.2.6.1.2.1.26.2.1.1

Table 79. ifMauTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
ifMauIfIndex	{ ifMauEntry 1 }	Integer32 (1…2147483647)	read-only	current	Y
ifMauIndex	{ ifMauEntry 2 }	Integer32 (1…2147483647)	read-only	current	Y
ifMauType	{ ifMauEntry 3 }	OBJECT IDENTIFIER	read-only	current	Y
ifMauStatus	{ ifMauEntry 4 }	INTEGER other (1) unknown (2) operational (3) standby (4) shutdown (5) reset (6)	read-only[a]	current	N
ifMauMediaAvailable	{ ifMauEntry 5 }	INTEGER other (1) unknown (2) available (3) notAvailable (4) remoteFault(5) invalidSignal(6) remoteJabber(7) remoteLinkLoss(8) remoteTest(9) offline(10) autoNegError(11)	read-only	current	N
ifMauMediaAvailableStateExits	{ ifMauEntry 6 }	Counter32	read-only	current	N
ifMauJabberState	{ ifMauEntry 7 }	INTEGER other(1) unknown(2) noJabber(3) jabbering(4)	read-only	current	N
ifMauJabberingStateEnters	{ ifMauEntry 8 }	Counter32	read-only	current	N
ifMauFalseCarriers	{ ifMauEntry 9 }	Counter32	read-only	current	N
ifMauTypeList	{ ifMauEntry 10 }	Integer32	read-only	deprecated	N
ifMauDefaultType	{ ifMauEntry 11 }	OBJECT IDENTIFIER	read-write	current	Y
ifMauAutoNegSupported	{ ifMauEntry 12 }	TruthValue	read-only	current	Y

Table 79. ifMauTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
ifMauTypeListBits	{ ifMauEntry 13 }	BITS bOther(0) bAUI(1) b10base5(2) bFoirl(3) b10base2(4) b10baseT(5) b10baseFP(6) b10baseFB(7) b10baseFL(8) b10broad36(9) b10baseTHD(10) b10baseTFD(11) b10baseFLHD(12) b10baseFLFD(13) b100baseT4(14) b100baseTXHD(15) b100baseTXFD(16) b100baseFXHD(17) b100baseFXFD(18) b100baseT2HD(19) b100baseT2FD(20) b1000baseXHD(21) b1000baseXFD(22) b1000baseLXHD(23) b1000baseLXFD(24) b1000baseSXHD(25) b1000baseSXFD(26) b1000baseCXHD(27) b1000baseCXFD(28) b1000baseTHD(29) b1000baseTFD(30)	read-only	current	N

a. Software control of the MAU is not required for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. As a result, this object is implemented as read-only.

5.9.4.2.1.1 ifMauIfIndex

This variable uniquely identifies the interface to which the MAU described by this entry is connected.

Valid Object Value(s)

This object will return the same value as the ifIndex associated with a particular Ethernet interface.

5.9.4.2.1.2 ifMauIndex

This variable uniquely identifies the MAU described by this entry from among other MAUs connected to the same interface (ifMauIfIndex).

Valid Object Value(s)

The concept of multiple MAUs on the same interface does not apply to the BitStorm 2600 and GranDSLAM 4200 IP DSLAM, so this value will always be “1.”

5.9.4.2.1.3 ifMauType

This object identifies the MAU type. An initial set of MAU types are defined above. The assignment of OBJECT IDENTIFIERs to new types of MAUs is managed by the IANA. If the MAU type is unknown, the object identifier unknownMauType OBJECT IDENTIFIER = { 0 0 } is returned. Note that unknownMauType is a syntactically valid object identifier, and any conformant implementation of ASN.1 and the BER must be able to generate and recognize this value.

This object represents the operational type of the MAU, as determined by either (1) the result of the auto-negotiation function or (2) if auto-negotiation is not enabled or is not implemented for this MAU, by the value of the object ifMauDefaultType. In case (2), a set to the object ifMauDefaultType will force the MAU into the new operating mode.

Valid Object Value(s)

The table below describes the MAU types and there associations with the interfaces applicable to the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

Table 80. ifMauType

Product Interface	Interface Type	Duplex Mode	MAU Type
Ethernet Port 1	10BASE-T	Half	dot3MauType10BaseTHD
		Full	dot3MauType10BaseTFD
	100BASE-TX	Half	dot3MauType100BaseTXHD
		Full	dot3MauType100BaseTXFD
Ethernet Port 2	10BASE-T	Half	dot3MauType10BaseTHD
		Full	dot3MauType10BaseTFD
	100BASE-TX	Half	dot3MauType100BaseTXHD
		Full	dot3MauType100BaseTXFD
Ethernet Port 3	10BASE-T	Half	dot3MauType10BaseTHD
		Full	dot3MauType10BaseTFD
	100BASE-TX	Half	dot3MauType100BaseTXHD
		Full	dot3MauType100BaseTXFD
	1000BASE-T	Half	dot3MauType1000BaseTHD
		Full	dot3MauType1000BaseTFD
	1000BASE-X[a]	Half	dot3MauType1000BaseXHD
		Full	dot3MauType1000BaseXFD

a. When a Small Form-factor Pluggable (SFP) module is installed, the MAU Type dot3MauType1000BaseX… identifies the SFP module.

5.9.4.2.1.4 ifMauStatus

This object is not supported.

The current state of the MAU. This object MAY be implemented as a read-only object by those agents and MAUs that do not implement software control of the MAU state. Some agents may not support setting the value of this object to some of the enumerated values. The following are values for this object:

- other(1) - The value other(1) is returned if the MAU is in a state other than one of the states 2 through 6.
- unknown(2) - The value unknown(2) is returned when the MAU's true state is unknown; for example, when it is being initialized.
- operational(3) - A MAU in the operational(3) state is fully functional, operates, and passes signals to its attached DTE or repeater port in accordance to its specification.
- standby(4) - A MAU in standby(4) state forces DI and CI to idle and the media transmitter to idle or fault, if supported. Standby(4) mode only applies to link type MAUs. The state of ifMauMediaAvailable is unaffected.
- shutdown(5) - A MAU in shutdown(5) state assumes the same condition on DI, CI, and the media transmitter as though it were powered down or not connected. The MAU MAY return other(1) value for the ifMauJabberState and ifMauMediaAvailable objects when it is in this state. For an AUI, this state will remove power from the AUI.
- reset(6) - Setting this variable to the value reset(6) resets the MAU in the same manner as a power-off, power-on cycle of at least one-half second would. The agent is not required to return the value reset (6).

Setting this variable to the value operational(3), standby(4), or shutdown(5) causes the MAU to assume the respective state except that setting a mixing-type MAU or an AUI to standby(4) will cause the MAU to enter the shutdown state.

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM is not required to control the MAU. This object is considered a status of the MAU and is implemented with an access scope of read-only.

5.9.4.2.1.5 ifMauMediaAvailable

This object is not supported.

If the MAU is a link or fiber type (FOIRL, 10BASE-T, 10BASE-F) then this is equivalent to the link test fail state/low light function. For an AUI or a coax (including broadband) MAU this indicates whether or not loopback is detected on the DI circuit. The value of this attribute persists between packets for MAU types AUI, 10BASE5, 10BASE2, 10BROAD36, and 10BASE-FP. The following are valid values for this object:

- other(1) - The value other(1) is returned if the mediaAvailable state is not one of 2 through 11.
- unknown(2) - The value unknown(2) is returned when the MAU's true state is unknown; for example, when it is being initialized. At power-up or following a reset, the value of this attribute is unknown for AUI, coax, and 10BASE-FP MAUs. For these MAUs loopback is tested on each transmission during which no collision is detected. If DI is receiving input when DO returns to IDL after a transmission, and there has been no collision during the transmission, then loopback will be detected. The value of this attribute will only change during non-collided transmissions for AUI, coax, and 10BASE-FP MAUs.

  For 100Mbps and 1000Mbps MAUs, the enumerations match the states within the respective link integrity state diagrams, fig 32-16, 23-12 and 24-15 of sections 32, 23 and 24 of [16]. Any MAU which implements management of auto-negotiation will map remote fault indication to remote fault.
- available(3) - The value available(3) indicates that the link, light, or loopback is normal.
- notAvailable(4) - The value notAvailable(4) indicates link loss, low light, or no loopback.

- remoteFault(5) - This value indicates that a fault has been detected at the remote end of the link. This value applies to 10BASE-FB, 100BASE-T4 Far End Fault Indication and non-specified remote faults from a system running auto-negotiation.
- invalidSignal(6) - This value indicates hat an invalid signal has been received from the other end of the link. InvalidSignal(6) applies only to MAUs of type 10BASE-FB.
- remoteJabber(7) - See explanation below.
- remoteLinkLoss(8) - See explanation below.
- remoteTest(9) - See explanation below.
- offline(10) - See explanation below.
- autoNegError(11) - See explanation below.

The values remoteJabber(7), remoteLinkLoss(8), and remoteTest(9) SHOULD be used instead of remoteFault(5) where the reason for remote fault is identified in the remote signaling protocol.

The value invalidSignal(6) indicates that an invalid signal has been received from the other end of the link. InvalidSignal(6) applies only to MAUs of type 10BASE-FB.

Where an IEEE Std 802.3u-1995 clause 22 MII is present, a logic one in the remote fault bit (reference section 22.2.4.2.8 of that document) maps to the value remoteFault(5), and a logic zero in the link status bit (reference section 22.2.4.2.10 of that document) maps to the value notAvailable(4). The value notAvailable(4) takes precedence over the value remoteFault(5).

Any MAU that implements management of clause 37 Auto-Negotiation will map the received RF1 and RF2 bit values for Offline to offline(10), Link Failure to remoteFault(5) and Auto-Negotiation Error to autoNegError(11).

5.9.4.2.1.6 ifMauMediaAvailableStateExits

This object is not supported.

A count of the number of times that ifMauMediaAvailable for this MAU instance leaves the state available(3). Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.9.4.2.1.7 ifMauJabberState

This object is not supported.

The following are valid values for this object:

- other(1) - The value other(1) is returned if the jabber state is not 2, 3, or 4. The agent MUST always return other(1) for MAU type dot3MauTypeAUI.
- unknown(2) - The value unknown(2) is returned when the MAU's true state is unknown; for example, when it is being initialized.
- noJabber(3) - If the MAU is not jabbering the agent returns noJabber(3). This is the “normal” state.
- jabbering(4) - If the MAU is in jabber state the agent returns the jabbering(4) value.

5.9.4.2.1.8 ifMauJabberingStateEnters

This object is not supported.

A count of the number of times that mauJabberState for this MAU instance enters the state jabbering(4). This counter will always indicate zero for MAUs of type dot1MauTypeAUI and those of speeds above 10Mbps.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.9.4.2.1.9 ifMauFalseCarriers

This object is not supported.

A count of the number of false carrier events during IDLE in 100BASE-X and 1000BASE-X links.

For all other MAU types, this counter will always indicate zero. This counter does not increment at the symbol rate.

It can increment after a valid carrier completion at a maximum rate of once per 100 ms for 100BASE-X and once per 10us for 1000BASE-X until the next CarrierEvent.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.

5.9.4.2.1.10 ifMauTypeList

This object is not supported.

THIS OBJECT IS DEPRECATED.

A value that uniquely identifies the set of possible IEEE 802.3 types that the MAU could be. The value is a sum which initially takes the value zero. Then, for each type capability of this MAU, 2 raised to the power noted below is added to the sum. For example, a MAU which has the capability to be only 10BASE-T would have a value of 512 (2**9). In contrast, a MAU which supports both 10Base-T (full duplex) and 100BASE-TX (full duplex) would have a value of ((2**11) + (2**16)) or 67584.

The powers of 2 assigned to the capabilities are these:

Table 81. ifMauTypeList

Power	Capability
0	other or unknown
1	reserved
2	AUI
3	10BASE-5
4	10BASE-2
5	10BASE-T duplex mode unknown
6	10BASE-FP
7	10BASE-FB
8	10BASE-FL duplex mode unknown
9	10BROAD36
10	10BASE-T half duplex mode
11	10BASE-T full duplex mode
12	10BASE-FL half duplex mode
13	10BASE-FL full duplex mode
14	100BASE-T4
15	100BASE-TX half duplex mode
16	100BASE-TX full duplex mode
17	00BASE-FX half duplex mode
18	100BASE-FX full duplex mode
19	100BASE-T2 half duplex mode
20	100BASE-T2 full duplex mode

If auto-negotiation is present on this MAU, this object will map to ifMauAutoNegCapability.

This object has been deprecated in favor of ifMauTypeListBits.

5.9.4.2.1.11 ifMauDefaultType

This object identifies the default administrative baseband MAU type, to be used in conjunction with the operational MAU type denoted by ifMauType.

The set of possible values for this object is the same as the set defined for the ifMauType object.

This object represents the administratively-configured type of the MAU. If auto-negotiation is not enabled or is not implemented for this MAU, the value of this object determines the operational type of the MAU. In this case, a set to this object will force the MAU into the specified operating mode.

If auto-negotiation is implemented and enabled for this MAU, the operational type of the MAU is determined by auto-negotiation, and the value of this object denotes the type to which the MAU will automatically revert if/when auto-negotiation is later disabled.

NOTE TO IMPLEMENTORS: It may be necessary to provide for underlying hardware implementations which do not follow the exact behavior specified above. In particular, when ifMauAutoNegAdminStatus transitions from enabled to disabled, the agent implementation MUST ensure that the operational type of the MAU (as reported by ifMauType) correctly transitions to the value specified by this object, rather than continuing to operate at the value earlier determined by the auto-negotiation function.

Valid Object Value(s)

The table below describes the default value for each of the interfaces applicable to the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. The essence of the table is that the interfaces will default to full duplex mode at their highest speed.

Table 82. ifMauDefaultType

Product Interface	Interface Type	Duplex Mode	MAU Type
Ethernet Port 1	10BASE-T 100BASE-TX	Full	dot3MauType100BaseTXHD
Ethernet Port 2	10BASE-T 100BASE-TX	Full	dot3MauType100BaseTXHD
Ethernet Port 3	10BASE-T 100BASE-TX 1000BASE-T	Full	dot3MauType1000BaseTHD
	1000BASE-FX[a]	Full	dot3MauType1000BaseXHD

a. This refers to the use of a Small Form-factor Pluggable (SFP) module. If the operator plugs in a SFP that operates at a speed that is less than 1000000 bps, it is incumbent on the operator to set the appropriate default MAU Type.

5.9.4.2.1.12 ifMauAutoNegSupported

This object indicates whether or not auto-negotiation is supported on this MAU.

This object will return a true(1) if auto-negotiation is supported, and false(2) otherwise.

5.9.4.2.1.13 ifMauTypeListBits

This object is not supported.

A value that uniquely identifies the set of possible IEEE 802.3 types that the MAU could be. If auto-negotiation is present on this MAU, this object will map to ifMauAutoNegCapability.

Note that this MAU may be capable of operating as a MAU type that is beyond the scope of this MIB. This is indicated by returning the bit value bOther in addition to any bit values for capabilities that are listed below. The bits below are those specified in the MIB's RFC. Those in bold font are possible values for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

- bOther(0) - other or unknown
- bAUI(1) - UI
- b10base5(2) - 10BASE-5
- bFoirl(3) - FOIRL
- b10base2(4) - 10BASE-2
- b10baseT(5) - 10BASE-T duplex mode unknown
- b10baseFP(6) - 10BASE-FP
- b10baseFB(7) - 10BASE-FB
- b10baseFL(8) - 10BASE-FL duplex mode unknown
- b10broad36(9) - 10BROAD36
- b10baseTHD(10) - 10BASE-T half duplex mode
- b10baseTFD(11) - 10BASE-T full duplex mode
- b10baseFLHD(12) - 10BASE-FL half duplex mode
- b10baseFLFD(13) - 10BASE-FL full duplex mode
- b100baseT4(14) - 100BASE-T4
- b100baseTXHD(15) - 100BASE-TX half duplex mode
- b100baseTXFD(16) - 100BASE-TX full duplex mode
- b100baseFXHD(17) - 100BASE-FX half duplex mode
- b100baseFXFD(18) - 100BASE-FX full duplex mode
- b100baseT2HD(19) - 100BASE-T2 half duplex mode
- b100baseT2FD(20) - 100BASE-T2 full duplex mode
- b1000baseXHD(21) - 1000BASE-X half duplex mode
- b1000baseXFD(22) - 1000BASE-X full duplex mode
- b1000baseLXHD(23) - 1000BASE-LX half duplex mode
- b1000baseLXFD(24) - 1000BASE-LX full duplex mode
- b1000baseSXHD(25) - 1000BASE-SX half duplex mode
- b1000baseSXFD(26) - 1000BASE-SX full duplex mode
- b1000baseCXHD(27) - 1000BASE-CX half duplex mode
- b1000baseCXFD(28) - 1000BASE-CX full duplex mode
- b1000baseTHD(29) - 1000BASE-T half duplex mode
- b1000baseTFD(30) - 1000BASE-T full duplex mode

5.9.4.2.2 ifJackTable

The ifJackTable applies to MAUs attached to interfaces which have one or more external jacks (connectors).

This table contains information about the external jacks attached to MAUs attached to an interface. This table is a sequence of ifJackEntry which contains information about a particular jack.

Indices:

- ifMauIfIndex (Section 5.9.4.2.1.1, Page 143)
- ifMauIndex (Section 5.9.4.2.1.2, Page 143)

- ifJackIndex

OIDs:

- ifJackTable - { dot3IfMauBasicGroup 2 } 1.2.6.1.2.1.26.2.2
- ifJackEntry - { ifJackTable 1 } 1.2.6.1.2.1.26.2.2.1

Table 83. ifJackTable

Object	OID	Syntax	Access	Status	Supported
ifJackIndex	{ ifJackEntry 1 }	Integer32 (1…2147483647)	not-accessible	current	Y
ifJackType	{ ifJackEntry 2 }	JackType	read-only	current	Y

5.9.4.2.2.1 ifJackIndex

This variable uniquely identifies the jack described by this entry from among other jacks attached to the same MAU.

This variable always returns a 1 since multiple jacks on a MAU does not apply to the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

5.9.4.2.2.2 ifJackType

The jack connector type, as it appears on the outside of the system.

The following table identifies the values for the ifJackType.

Table 84. ifJackType

Product Interface	Interface Type	Jack[a]	Return Value
Ethernet Port 1	10BASE-T	Eight Pin Modular	rj45(2)
	100BASE-TX		
Ethernet Port 2	10BASE-T	Eight Pin Modular	rj45(2)
	100BASE-TX		
Ethernet Port 3	10BASE-T	Eight Pin Modular	rj45(2)
	100BASE-TX		
	1000BASE-T		
	100BASE-FX	Small Form-factor Pluggable (SFP)[b]	other(1)
	1000BASE-SX		
	1000BASE-LX		

a. Although an eight pin modular jack is not necessarily an RJ45-type jack, they are equated here.

b. When the SFP is installed, it takes precedence over the Eight Pin Modular.

5.9.4.3 Interface MAU Auto-Negotiation Group

The ifMauAutoNegTable applies to systems in which auto-negotiation is supported on one or more MAUs attached to interfaces. Note that if auto-negotiation is present and enabled, the ifMauType object reflects the result of the auto-negotiation function.

5.9.4.3.1 ifMauAutoNegTable

This table contains configuration and status objects for the auto-negotiation function of MAUs attached to interfaces. The table is a sequence of ifMauAutoNegEntry which contains configuration and status information for the auto-negotiation function of a particular MAU.

Indices:

- ifMauIfIndex (Section 5.9.4.2.1.1, Page 143)
- ifMauIndex (Section 5.9.4.2.1.2, Page 143)

OIDs:

- ifMauAutoNegTable - { dot3IfMauAutoNegGroup 1 } 1.2.6.1.2.1.26.5.1
- ifMauAutoNegEntry - { ifMauAutoNegTable 1 } 1.2.6.1.2.1.26.5.1.1

Table 85. ifMauAutoNegTable (1 of 3)

Object	OID	Syntax	Access	Status	Supported
ifMauAutoNegAdminStatus	{ ifMauAutoNegEntry 1 }	INTEGER enabled(1) disabled(2)	read-write	current	Y
ifMauAutoNegRemoteSignaling	{ ifMauAutoNegEntry 2 }	INTEGER detected(1) notdetected(2)	read-only	current	N
ifMauAutoNegConfig	{ ifMauAutoNegEntry 4 }	INTEGER other(1) configuring(2) complete(3) disabled(4) parallelDetectFail(5)	read-only	current	N
ifMauAutoNegCapability	{ ifMauAutoNegEntry 5 }	Integer32	read-only	deprecated	N
ifMauAutoNegCapAdvertised	{ ifMauAutoNegEntry 6 }	Integer32	read-write	deprecated	N
ifMauAutoNegCapReceived	{ ifMauAutoNegEntry 7 }	Integer32	read-only	deprecated	N
ifMauAutoNegRestart	{ ifMauAutoNegEntry 8 }	INTEGER restert(1) norestart(2)	read-write	current	N

Table 85. ifMauAutoNegTable (2 of 3)

Object	OID	Syntax	Access	Status	Supported
ifMauAutoNegCapabilityBits	{ ifMauAutoNegEntry 9 }	BITS bother(0) b10baseT(1) b10baseTFD(2) b100baseT4(3) b100baseTX(4) b100baseTXFD(5) b100baseT2(6) b100baseT2FD(7) bFdxPause(8) bFdxAPause(9) bFdxSPause(10) bFdxBPause(11) b1000baseX(12) b1000baseXFD(13) b1000baseT(14) b1000baseTFD(15)	read-only	current	N
ifMauAutoNegCapAdvertisedBits	{ ifMauAutoNegEntry 10 }	BITS bother(0) b10baseT(1) b10baseTFD(2) b100baseT4(3) b100baseTX(4) b100baseTXFD(5) b100baseT2(6) b100baseT2FD(7) bFdxPause(8) bFdxAPause(9) bFdxSPause(10) bFdxBPause(11) b1000baseX(12) b1000baseXFD(13) b1000baseT(14) b1000baseTFD(15)	read-write	current	N

Table 85. ifMauAutoNegTable (3 of 3)

Object	OID	Syntax	Access	Status	Supported
ifMauAutoNegCapReceivedBits	{ ifMauAutoNegEntry 11 }	BITS bother(0) b10baseT(1) b10baseTFD(2) b100baseT4(3) b100baseTX(4) b100baseTXFD(5) b100baseT2(6) b100baseT2FD(7) bFdxPause(8) bFdxAPause(9) bFdxSPause(10) bFdxBPause(11) b1000baseX(12) b1000baseXFD(13) b1000baseT(14) b1000baseTFD(15)	read-only	current	N
ifMauAutoNegRemoteFaultAdvertised	{ ifMauAutoNegEntry 12 }	INTEGER noError(1) offline(2) linkFailure(3) autoNegError(4)	read-write	current	N
ifMauAutoNegRemoteFaultReceived	{ ifMauAutoNegEntry 13 }	INTEGER noError(1) offline(2) linkFailure(3) autoNegError(4)	read-only	current	N

5.9.4.3.1.1 ifMauAutoNegAdminStatus

- enabled(1) - If the interface has auto-negotiation signaling ability, auto-negotiation is enabled.
- disabled(2) - The interface will act as it would if it had no auto-negotiation signaling. Under these conditions, an IEEE 802.3 MAU will immediately be forced to the state indicated by the value of the object ifMauDefaultType

NOTE TO IMPLEMENTORS: When ifMauAutoNegAdminStatus transitions from enabled to disabled, the agent implementation MUST ensure that the operational type of the MAU (as reported by ifMauType) correctly transitions to the value specified by the ifMauDefaultType object, rather than continuing to operate at the value earlier determined by the auto-negotiation function.

Valid Object Value(s)

- Internal Ethernet Interfaces
  — disabled(2)
- External Ethernet Interfaces
  — enabled(1)

— disabled(2)

5.9.4.3.1.2 ifMauAutoNegRemoteSignaling

This object is not supported.

A value indicating whether the remote end of the link is using auto-negotiation signaling.

- detected(1) - It takes the value detected(1) if and only if, during the previous link negotiation, FLP Bursts were received.
- notdetected(2)

5.9.4.3.1.3 ifMauAutoNegConfig

This object is not supported.

A value indicating the current status of the auto-negotiation process. The enumeration parallelDetectFail(5) maps to a failure in parallel detection as defined in 28.2.3.1 of [IEEE 802.3 Std].

- other(1)
- configuring(2)
- complete(3)
- disabled(4)
- parallelDetectFail(5)

5.9.4.3.1.4 ifMauAutoNegCapability

This object is not supported.

THIS OBJECT IS DEPRECATED.

A value that uniquely identifies the set of capabilities of the local auto-negotiation entity. The value is a sum which initially takes the value zero. Then, for each capability of this interface, 2 raised to the power noted below is added to the sum. For example, an interface which has the capability to support only 100Base-TX half duplex would have a value of 32768 (2**15). In contrast, an interface which supports both 100Base-TX half duplex and 100Base-TX full duplex would have a value of 98304 ((2**15) + (2**16)).

The powers of 2 assigned to the capabilities are these:

Table 86. ifMauAutoNegCapability (1 of 2)

Power	Capability
0	other or unknown
1-9	reserved
10	10BASE-T half duplex mode
11	10BASE-T full duplex mode
12	reserved
13	reserved
14	100BASE-T4
15	100BASE-TX half duplex mode
16	100BASE-TX full duplex mode
17	reserved
18	reserved

Table 86. ifMauAutoNegCapability (2 of 2)

Power	Capability
19	100BASE-T2 half duplex mode
20	100BASE-T2 full duplex mode

Note that interfaces that support this MIB may have capabilities that extend beyond the scope of this MIB. This object has been deprecated in favor of ifMauAutoNegCapabilityBits

5.9.4.3.1.5 ifMauAutoNegCapAdvertised

This object is not supported.

THIS OBJECT IS DEPRECATED.

A value that uniquely identifies the set of capabilities advertised by the local auto-negotiation entity. Refer to ifMauAutoNegCapability for a description of the possible values of this object.

Capabilities in this object that are not available in ifMauAutoNegCapability cannot be enabled.

This object has been deprecated in favor of ifMauAutoNegCapAdvertisedBits

5.9.4.3.1.6 ifMauAutoNegCapReceived

This object is not supported.

THIS OBJECT IS DEPRECATED.

A value that uniquely identifies the set of capabilities received from the remote auto-negotiation entity. Refer to ifMauAutoNegCapability for a description of the possible values of this object.

Note that interfaces that support this MIB may be attached to remote auto-negotiation entities which have capabilities beyond the scope of this MIB.

This object has been deprecated in favor of ifMauAutoNegCapReceivedBits

5.9.4.3.1.7 ifMauAutoNegRestart

This object is not supported.

- restart(1) - Force auto-negotiation to begin link renegotiation. If auto-negotiation signaling is disabled, a write to this object has no effect.
- norestart(2) - Setting the value of this object to norestart(2) has no effect

5.9.4.3.1.8 ifMauAutoNegCapabilityBits

This object is not supported.

A value that uniquely identifies the set of capabilities of the local auto-negotiation entity. Note that interfaces that support this MIB may have capabilities that extend beyond the scope of this MIB.

Note that the local auto-negotiation entity may support some capabilities beyond the scope of this MIB. This is indicated by returning the bit value bOther in addition to any bit values for capabilities that are listed below.

- bOther(0) - other or unknown
- b10baseT(1) - 10BASE-T half duplex mode
- b10baseTFD(2) - 10BASE-T full duplex mode
- b100baseT4(3) - 100BASE-T4
- b100baseTX(4) - 100BASE-TX half duplex mode
- b100baseTXFD(5) - 100BASE-TX full duplex mode
- b100baseT2(6) - 100BASE-T2 half duplex mode

- b100baseT2FD(7) - 100BASE-T2 full duplex mode
- bfdxPause(8) - PAUSE for full-duplex links
- bfdxAPause(9) - Asymmetric PAUSE for full-duplex links
- bfdxSPause(10) - Symmetric PAUSE for full-duplex links
- bfdxBPause(11) - Asymmetric and Symmetric PAUSE for full-duplex links
- b1000baseX(12) - 1000BASE-X, -LX, -SX, -CX half duplex mode
- b1000baseXFD(13) - 1000BASE-X, -LX, -SX, -CX full duplex mode
- b1000baseT(14) - 1000BASE-T half duplex mode
- b1000baseTFD(15) - 1000BASE-T full duplex mode

5.9.4.3.1.9 ifMauAutoNegCapAdvertisedBits

This object is not supported.

A value that uniquely identifies the set of capabilities advertised by the local auto-negotiation entity.

Capabilities in this object that are not available in ifMauAutoNegCapabilityBits cannot be enabled. Note that the local auto-negotiation entity may advertise some capabilities beyond the scope of this MIB. This is indicated by returning the bit value bOther in addition to any bit values for capabilities that are listed below.

- bOther(0) - other or unknown
- b10baseT(1) - 10BASE-T half duplex mode
- b10baseTFD(2) - 10BASE-T full duplex mode
- b100baseT4(3) - 100BASE-T4
- b100baseTX(4) - 100BASE-TX half duplex mode
- b100baseTXFD(5) - 100BASE-TX full duplex mode
- b100baseT2(6) - 100BASE-T2 half duplex mode
- b100baseT2FD(7) - 100BASE-T2 full duplex mode
- bfdxPause(8) - PAUSE for full-duplex links
- bfdxAPause(9) - Asymmetric PAUSE for full-duplex links
- bfdxSPause(10) - Symmetric PAUSE for full-duplex links
- bfdxBPause(11) - Asymmetric and Symmetric PAUSE for full-duplex links
- b1000baseX(12) - 1000BASE-X, -LX, -SX, -CX half duplex mode
- b1000baseXFD(13) - 000BASE-X, -LX, -SX, -CX full duplex mode
- b1000baseT(14) - 1000BASE-T half duplex mode
- b1000baseTFD(15) - 1000BASE-T full duplex mode

5.9.4.3.1.10 ifMauAutoNegCapReceivedBits

This object is not supported.

A value that uniquely identifies the set of capabilities received from the remote auto-negotiation entity.

Note that interfaces that support this MIB may be attached to remote auto-negotiation entities which have capabilities beyond the scope of this MIB. This is indicated by returning the bit value bOther in addition to any bit values for capabilities that are listed below.

- bOther(0) - other or unknown
- b10baseT(1) - 10BASE-T half duplex mode
- b10baseTFD(2) - 10BASE-T full duplex mode

- b100baseT4(3) - 100BASE-T4
- b100baseTX(4) - 100BASE-TX half duplex mode
- b100baseTXFD(5) - 100BASE-TX full duplex mode
- b100baseT2(6) - 100BASE-T2 half duplex mode
- b100baseT2FD(7) - 100BASE-T2 full duplex mode
- bfdxPause(8) - PAUSE for full-duplex links
- bfdxAPause(9) - Asymmetric PAUSE for full-duplex links
- bfdxSPause(10) - Symmetric PAUSE for full-duplex links
- bfdxBPause(11) - Asymmetric and Symmetric PAUSE for full-duplex links
- b1000baseX(12) - 1000BASE-X, -LX, -SX, -CX half duplex mode
- b1000baseXFD(13) - 000BASE-X, -LX, -SX, -CX full duplex mode
- b1000baseT(14) - 1000BASE-T half duplex mode
- b1000baseTFD(15) - 1000BASE-T full duplex mode

5.9.4.3.1.11 ifMauAutoNegRemoteFaultAdvertised

This object is not supported.

A value that identifies any local fault indications that this MAU has detected and will advertise at the next auto-negotiation interaction for 1000Mbps MAUs.

- noError(1)
- offline(2)
- linkFailure(3)
- autoNegError(4)

5.9.4.3.1.12 ifMauAutoNegRemoteFaultReceived

This object is not supported.

A value that identifies any fault indications received from the far end of a link by the local auto-negotiation entity for 1000Mbps MAUs.

- noError(1)
- offline(2)
- linkFailure(3)
- autoNegError(4)

5.9.4.4 dot3BroadMauBasicGroup

5.9.4.4.1 broadMauBasicTable

THIS OBJECT IS DEPRECATED.

Table of descriptive and status information about the broadband MAUs connected to interfaces. This table is a sequence of broadMauBasicEntry which contains information about a single broadband MAU.

This deprecated table will not be supported in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

Index:

- broadMauIfIndex

OIDs:

- broadMauBasicTable - { dot3BroadMauBasicGroup 1 } 1.2.6.1.2.1.26.3.1
- broadMauBasicEntry - { broadMauBasicTable 1 } 1.2.6.1.2.1.26.3.1.1

Table 87. broadMauBasicTable

Object	OID	Syntax	Access	Status	Supported
broadMauIfIndex	{ broadMauBasicEntry 1 }	Integer32 (1…2147483647)	read-only	deprecated	N
broadMauIndex	{ broadMauBasicEntry 2 }	Integer32 (1…2147483647)	read-only	deprecated	N
broadMauXmtRcvSplitType	{ broadMauBasicEntry 3 }	INTEGER other(1) single(2) dual(3)	read-only	deprecated	N
broadMauXmtCarrierFreq	{ broadMauBasicEntry 4 }	Integer32	read-only	deprecated	N
broadMauTranslationFreq	{ broadMauBasicEntry 5 }	Integer32	read-only	deprecated	N

5.9.4.4.1.1 broadMauIfIndex

THIS OBJECT IS DEPRECATED.

This variable uniquely identifies the interface to which the MAU described by this entry is connected.

5.9.4.4.1.2 broadMauIndex

THIS OBJECT IS DEPRECATED.

This variable uniquely identifies the MAU connected to interface broadMauIfIndex that is described by this entry.

5.9.4.4.1.3 broadMauXmtRcvSplitType

THIS OBJECT IS DEPRECATED.

This object indicates the type of frequency multiplexing/cabling system used to separate the transmit and receive paths for the 10BROAD36 MAU.

- other(1) - The split type is not either single or dual.
- single(2) - Indicates a single cable system
- dual(3) - Indicates a dual cable system, offset normally zero.

5.9.4.4.1.4 broadMauXmtCarrierFreq

THIS OBJECT IS DEPRECATED.

This variable indicates the transmit carrier frequency of the 10BROAD36 MAU in MHz/4; that is, in units of 250 kHz.

5.9.4.4.1.5 broadMauTranslationFreq

THIS OBJECT IS DEPRECATED.

This variable indicates the translation offset frequency of the 10BROAD36 MAU in MHz/4; that is, in units of 250 kHz.

5.9.5 Traps

No 802.3 MAU Traps are supported.

OIDs:

- snmpDot3MauTraps - { snmpDot3MauMgt 0 } 1.2.6.1.2.1.26.0

Table 88. 802.3 MAU Traps

Notification	OID	Variable Bindings	Supported
rpMauJabberTrap	{ snmpDot3MauTraps 1 }	rpMauJabberState	N
ifMauJabberTrap	{ snmpDot3MauTraps 2 }	ifMauIfIndex ifMauIndex ifMauJabberState	N

5.9.5.1 rpMauJabberTrap

This trap is sent whenever a managed repeater MAU enters the jabber state.

The agent MUST throttle the generation of consecutive rpMauJabberTraps so that there is at least a five-second gap between them.

5.9.5.2 ifMauJabberTrap

This trap is sent whenever a managed interface MAU enters the jabber state.

The agent MUST throttle the generation of consecutive ifMauJabberTraps so that there is at least a five-second gap between them.

5.10 RS-232-MIB — RFC 1659

5.10.1 Introduction

This MIB describes objects for managing RS-232-like interfaces. The RFC used as a reference was dated July 1994. The LAST-UPDATED for the RS-232-MIB was "9405261700Z."

This MIB applies to the following interfaces in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM:

- Console

5.10.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- transmission - { mib-2 10 } 1.3.6.1.2.1.10
- rs232 - { transmission 33 } 1.3.6.1.2.1.10.33

5.10.3 MIB Objects

This MIB comprises one base object and four tables, detailed in the following sections. The tables contain objects for all ports, asynchronous ports, and input and output control signals.

Table 89. RS-232-MIB Objects

Object	OID	Syntax	Access	Status	Supported
rs232Number	{ rs232 1 }	Integer32	read-only	current	Y
rs232PortTable	{ rs232 2 }	SEQUENCE of rs232PortEntry	not-accessible	current	Y
rs232AsyncPortTable	{ rs232 3 }	SEQUENCE of rs232AyncPortEntry	not-accessible	current	Y
rs232SyncPortTable	{ rs232 4 }	SEQUENCE of rs232SyncPortEntry	not-accessible	current	N
rs232InSigTable	{ rs232 5 }	SEQUENCE of rs232InSigEntry	not-accessible	current	N
rs232OutSigTable	{ rs232 6 }	SEQUENCE of rs232OutSigEntry	not-accessible	current	N

5.10.3.1 rs232Number

The number of ports (regardless of their current state) in the RS-232-like general port table.

Valid Object Value(s)

- 1

5.10.3.2 rs232PortTable

A list of port entries. The number of entries is given by the value of rs232Number. This table is a sequence of rs232PortEntry which contains Status and parameter values for a port.

Index:

- rs232PortIndex

OIDs:

- rs232PortTable - { rs232 2 } 1.3.6.1.2.1.10.33.2

- rs232PortEntry - { rs232PortTable 1} 1.3.6.1.2.1.10.33.2.1

Table 90. rs232PortTable

Object	OID	Syntax	Access	Status	Supported
rs232PortIndex	{ rs232PortEntry 1 }	InterfaceIndex	read-only	current	Y
rs232PortType	{ rs232PortEntry 2 }	INTEGER other(1) rs232(2) rs422(3) rs423(4) v35(5) x21(6)	read-only	current	Y
rs232PortInSigNumber	{ rs232PortEntry 3 }	Integer32	read-only	current	Y
rs232PortOutSigNumber	{ rs232PortEntry 4 }	Integer32	read-only	current	Y
rs232PortInSpeed	{ rs232PortEntry 5 }	Integer32	read-write	current	Y
rs232PortOutSpeed	{ rs232PortEntry 6 }	Integer32	read-write	current	Y
rs232PortInFlowType	{ rs232PortEntry 7 }	INTEGER none(1) ctsRts(2) dsrDtr(3)	read-write	current	Y
rs232PortOutFlowType	{ rs232PortEntry 8 }	INTEGER none(1) ctsRts(2) dsrDtr(3)	read-write	current	Y

5.10.3.2.1 rs232PortIndex

The value of ifIndex for the port. By convention and if possible, hardware port numbers map directly to external connectors. The value for each port must remain constant at least from one re-initialization of the network management agent to the next.

Valid Object Value(s)

This object is equated to the ifIndex (Section 5.3.4.3.1, Page 37).

5.10.3.2.2 rs232PortType

The port's hardware type.

- other(1)
- rs232(2)
- rs422(3)
- rs423(4)
- v35(5)
- x21(6)

Valid Object Value(s)

- Console: rs232(2)

5.10.3.2.3 rs232PortInSigNumber

The number of input signals for the port in the input signal table (rs232InSigTable, Section 5.10.3.5, Page 168). The table contains entries only for those signals the software can detect and that are useful to observe.

5.10.3.2.4 rs232PortOutSigNumber

The number of output signals for the port in the output signal table (rs232OutSigTable, Section 5.10.3.6, Page 170). The table contains entries only for those signals the software can assert and that are useful to observe.

5.10.3.2.5 rs232PortInSpeed

The port's input speed in bits per second. Note that non-standard values, such as 9612, are probably not allowed on most implementations.

Valid Object Value(s)

Note: Port speeds are symmetrical. Therefore the rs232PortInSpeed object is the same as the rs232PortOutSpeed object. From the standpoint of the internals, when you set one, internally the other gets set to the same value as well.

- Console: 9600, 19200, 38400, 57600, 115200 bps

5.10.3.2.6 rs232PortOutSpeed

The port's output speed in bits per second. Note that non-standard values, such as 9612, are probably not allowed on most implementations.

Valid Object Value(s)

See rs232PortInSpeed above.

5.10.3.2.7 rs232PortInFlowType

The port's type of input flow control. “none” indicates no flow control at this level. “ctsRts” and “dsrDtr” indicate use of the indicated hardware signals.

- none(1)
- ctsRts(2)
- dsrDtr(3)

This object is interpreted from the perspective of the port. Input flow control indicates which output signal gets set to false to signal the device attached to the other side of the communication link to stop transmitting data. This port controls the flow of the remote device.

Valid Object Value(s)

The only valid value for this object is none(1).

5.10.3.2.8 rs232PortOutFlowType

The port's type of output flow control. “none” indicates no flow control at this level. “ctsRts” and “dsrDtr” indicate use of the indicated hardware signals.

- none(1)
- ctsRts(2)
- dsrDtr(3)

This object is interpreted from the perspective of the port. Output flow control indicates which input signal to monitor such that when it set to false, this port should stop transmitting data to the other side of the communication link. The remote device controls the flow of this port.

Valid Object Value(s)

- Console: none(1)

5.10.3.3 rs232AsyncPortTable

A list of asynchronous port entries. Entries need not exist for synchronous ports. This table is a sequence of rs232AsyncPortEntry which contains status and parameter values for an asynchronous port.

The following interfaces on the BitStorm 2600 and GranDSLAM 4200 IP DSLAM apply to the rs232AsyncPort type:

- Console

Index:

- rs232AsyncPortIndex

OIDs:

- rs232AsyncPortTable - { rs232 3 } 1.3.6.1.2.1.10.33.3
- rs232AsyncPortEntry - { rs232AsyncPortTable 1} 1.3.6.1.2.1.10.33.3.1

Table 91. rs232AsyncPortTable

Object	OID	Syntax	Access	Status	Supported
rs232AsyncPortIndex	{ rs232AsyncPortEntry 1 }	InterfaceIndex	read-only	current	Y
rs232AsyncPortBits	{ rs232AsyncPortEntry 2 }	INTEGER(5…8)	read-write	current	Y
rs232AsyncPortStopBits	{ rs232AsyncPortEntry 3 }	INTEGER one(1) two(2) oneAndHalf(3) dynamic(4)	read-write	current	Y
rs232AsyncPortParity	{ rs232AsyncPortEntry 4 }	INTEGER none(1) odd(2) even(3) mark(4) space(5)	read-write	current	Y
rs232AsyncPortAutobaud	{ rs232AsyncPortEntry 5 }	INTEGER enabled(1) disabled(2)	read-write	current	Y
rs232AsyncPortParityErrs	{ rs232AsyncPortEntry 6 }	Counter32	read-only	current	N
rs232AsyncPortFramingErrs	{ rs232AsyncPortEntry 7 }	Counter32	read-only	current	N
rs232AsyncPortOverrunErrs	{ rs232AsyncPortEntry 8 }	Counter32	read-only	current	N

5.10.3.3.1 rs232AsyncPortIndex

A unique value for each port. Its value is the same as rs232PortIndex, Section 5.10.3.2.1, Page 162, for the port.

5.10.3.3.2 rs232AsyncPortBits

The port's number of bits in a character.

Valid Object Value(s)

- Console: 7 or 8

5.10.3.3.3 rs232AsyncPortStopBits

The port's number of stop bits.

- one(1)
- two(2)
- oneAndHalf(3)
- dynamic(4)

Valid Object Value(s)

- Console: one(1) or two(2)

5.10.3.3.4 rs232AsyncPortParity

The port's sense of a character parity bit.

- none(1)
- odd(2)
- even(3)
- mark(4)
- space(5)

Valid Object Value(s)

- Console: none(1), odd(2), even(3)

5.10.3.3.5 rs232AsyncPortAutobaud

A control for the port's ability to automatically sense input speed. When rs232PortAutoBaud is “enabled,” a port may autobaud to values different from the set values for speed, parity, and character size. As a result a network management system may temporarily observe values different from what was previously set.

- enabled(1)
- disabled(2)

Valid Object Value(s)

- Console: disabled(2)

5.10.3.3.6 rs232AsyncPortParityErrs

Total number of characters with a parity error, input from the port since system re-initialization and while the port state was “up” or “test.”

Valid Object Value(s)

This object will not be supported.

5.10.3.3.7 rs232AsyncPortFramingErrs

Total number of characters with a framing error, input from the port since system re-initialization and while the port state was “up” or “test.”

Valid Object Value(s)

This object will not be supported.

5.10.3.3.8 rs232AsyncPortOverrunErrs

Total number of characters with an overrun error, input from the port since system re-initialization and while the port state was “up” or “test.”

Valid Object Value(s)

This object will not be supported.

5.10.3.4 rs232SyncPortTable

A list of asynchronous port entries. Entries need not exist for synchronous ports. This table is a sequence of rs232SyncPortEntry where each entry consists of status and parameter values for a synchronous port.

Index:

- rs232SyncPortIndex

OIDs:

- rs232SyncPortTable - { rs232 4 } 1.3.6.1.2.1.10.33.4
- rs232SyncPortEntry - { rs232SyncPortTable 1} 1.3.6.1.2.1.10.33.4.1

Table 92. rs232SyncPortTable

Object	OID	Syntax	Access	Status	Supported
rs232SyncPortIndex	{ rs232SyncPortEntry 1 }	InterfaceIndex	read-only	current	N
rs232SyncPortClockSource	{ rs232SyncPortEntry 2 }	INTEGER internal(1) external(2) split(3)	read-write	current	N
rs232SyncPortFrameCheckErrs	{ rs232SyncPortEntry 3 }	Counter32	read-only	current	N
rs232SyncPortTransmitUnderrunErrs	{ rs232SyncPortEntry 4 }	Counter32	read-only	current	N
rs232SyncPortReceiveOverrunErrs	{ rs232SyncPortEntry 5 }	Counter32	read-only	current	N
rs232SyncPortInterruptedFrames	{ rs232SyncPortEntry 6 }	Counter32	read-only	current	N
rs232SyncPortAbortedFrames	{ rs232SyncPortEntry 7 }	Counter32	read-only	current	N
rs232SyncPortRole	{ rs232SyncPortEntry 8 }	INTEGER dte(1) dce(2)	read-write	current	N
rs232SyncPortEncoding	{ rs232SyncPortEntry 9 }	INTEGER nrz(1) nrzi(2)	read-write	current	N
rs232SyncPortRTSControl	{ rs232SyncPortEntry 10 }	INTEGER controlled(1) constant(2)	read-write	current	N
rs232SyncPortRTSCTSDelay	{ rs232SyncPortEntry 11 }	Integer32	read-write	current	N
rs232SyncPortMode	{ rs232SyncPortEntry 12 }	INTEGER fdx(1) hdx(2) simplex-received(3) simplex-send(4)	read-write	current	N

Table 92. rs232SyncPortTable

Object	OID	Syntax	Access	Status	Supported
rs232SyncPortIdlePattern	{ rs232SyncPortEntry 13 }	INTEGER mark(1) space(2)	read-write	current	N
rs232SyncPortMinFlags	{ rs232SyncPortEntry 14 }	Integer32	read-write	current	N

5.10.3.4.1 rs232SyncPortIndex

A unique value for each port. Its value is the same as rs232PortIndex, Section 5.10.3.2.1, Page 162, for the port.

5.10.3.4.2 rs232SyncPortClockSource

Source of the port's bit rate clock. “split”" means the transmit clock is internal and the receive clock is external.

- internal(1)
- external(2)
- split(3)

5.10.3.4.3 rs232SyncPortFrameCheckErrs

Total number of frames with an invalid frame check sequence, input from the port since system re-initialization and while the port state was “up” or “test.”

5.10.3.4.4 rs232SyncPortTransmitUnderrunErrs

Total number of frames that failed to be transmitted on the port since system re-initialization and while the port state was “up or test” because data was not available to the transmitter in time.

5.10.3.4.5 rs232SyncPortReceiveOverrunErrs

Total number of frames that failed to be received on the port since system re-initialization and while the port state was “up or test” because the receiver did not accept the data in time.

5.10.3.4.6 rs232SyncPortInterruptedFrames

Total number of frames that failed to be received or transmitted on the port due to loss of modem signals since system re-initialization and while the port state was “up” or “test.”

5.10.3.4.7 rs232SyncPortAbortedFrames

Number of frames aborted on the port due to receiving an abort sequence since system re-initialization and while the port state was “up” or “test.”

5.10.3.4.8 rs232SyncPortRole

The role the device is playing that is using this port. The default value is dce(2).

- dte(1) - means the device is performing the role of data terminal equipment
- dce(2) - means the device is performing the role of data circuit-terminating equipment.

5.10.3.4.9 rs232SyncPortEncoding

The bit stream encoding technique that is in effect for this port.

- nrz(1) - for Non-Return to Zero encoding
- nrzi(2) - for Non-Return to Zero Inverted encoding.

5.10.3.4.10 rs232SyncPortRTSControl

The method used to control the Request To Send (RTS) signal. The default value is constant(2).

- controlled(1) - when the DTE is asserts RTS each time data needs to be transmitted and drops RTS at some point after data transmission begins.

  If rs232SyncPortRole is “dte,” the RTS is an output signal. The device will issue a RTS and wait for a CTS from the DCE before starting to transmit.

  If rs232SyncPortRole is “dce,” the RTS is an input signal. The device will issue a CTS only after having received RTS and waiting the rs232SyncPortRTSCTSDelay interval.
- constant(2) - when the DTE constantly asserts RTS.

5.10.3.4.11 rs232SyncPortRTSCTSDelay

The interval (in milliseconds) that the DCE must wait after it sees RTS asserted before asserting CTS. This object exists in support of older synchronous devices that cannot recognize CTS within a certain interval after it asserts RTS.

5.10.3.4.12 rs232SyncPortMode

The mode of operation of the port with respect to the direction and simultaneity of data transfer. The default value is fdx.

- fdx(1) - when frames on the data link can be transmitted and received at the same time
- hdx(2) - when frames can either be received from the data link or transmitted onto the data link but not at the same time.
- simplex-receive(3) - when frames can only be received on this data link.
- simplex-send(4) - when frames can only be sent on this data link.

5.10.3.4.13 rs232SyncPortIdlePattern

The bit pattern used to indicate an idle line. The default is space(2).

- mark(1)
- space(2)

5.10.3.4.14 rs232SyncPortMinFlags

The minimum number of flag patterns this port needs in order to recognize the end of one frame and the start of the next. Plausible values are 1 and 2. The default value is “2.”

5.10.3.5 rs232InSigTable

A list of port input control signal entries implemented and visible to the software on the port, and useful to monitor. This table is a sequence of rs232InSigEntry and each entry consists of input control signal status for a hardware port.

Indices:

- rs232InSigPortIndex
- rs232InSigName

OIDs:

- rs232InSigTable - { rs232 5 } 1.3.6.1.2.1.10.33.5
- rs232InSigEntry - { rs232InSigTable 1} 1.3.6.1.2.1.10.33.5.1

Table 93. rs232InSigTable

Object	OID	Syntax	Access	Status	Supported
rs232InSigPortIndex	{ rs232InSigEntry 1 }	InterfaceIndex	read-only	current	N
rs232InSigName	{ rs232InSigEntry 2 }	INTEGER rts1) cts(2) dsr(3) dtr(4) ri(5) dcd(6) sq(7) srs(8) srts(9) scts(10) sdcd(11)	read-only	current	N
rs232InSigState	{ rs232InSigEntry 3 }	INTEGER none(1) on(2) off(3)	read-only	current	N
rs232InSigChanges	{ rs232InSigEntry 4 }	Counter32	read-only	current	N

5.10.3.5.1 rs232InSigPortIndex

The value of rs232PortIndex, Section 5.10.3.2.1, Page 162, for the port to which this entry belongs.

5.10.3.5.2 rs232InSigName

Identification of a hardware signal, as follows:

- rts - Request to Send
- cts - Clear to Send
- dsr - Data Set Ready
- dtr - Data Terminal Ready
- ri - Ring Indicator
- dcd - Received Line Signal Detector
- sq - Signal Quality Detector
- srs - Data Signaling Rate Selector
- srts - Secondary Request to Send
- scts - Secondary Clear to Send
- sdcd - Secondary Received Line Signal Detector

5.10.3.5.3 rs232InSigState

The current signal state.

- none(1)

- on(2)
- off(3)

5.10.3.5.4 rs232InSigChanges

The number of times the signal has changed from “on” to “off” or from “off” to “on.”

5.10.3.6 rs232OutSigTable

A list of port output control signal entries implemented and visible to the software on the port, and useful to monitor. This table is a sequence of rs232OutSigEntry where each entry consists of Output control signal status for a hardware port.

Indices:

- rs232OutSigPortIndex
- rs232OutSigName

OIDs:

- rs232OutSigTable - { rs232 6 } 1.3.6.1.2.1.10.33.6
- rs232OutSigEntry - { rs232OutSigTable 1} 1.3.6.1.2.1.10.33.6.1

Table 94. rs232OutSigTable

Object	OID	Syntax	Access	Status	Supported
rs232OutSigPortIndex	{ rs232OutSigEntry 1 }	InterfaceIndex	read-only	current	N
rs232OutSigName	{ rs232OutSigEntry 2 }	INTEGER rts1) cts(2) dsr(3) dtr(4) ri(5) dcd(6) sq(7) srs(8) srts(9) scts(10) sdcd(11)	read-only	current	N
rs232OutSigState	{ rs232OutSigEntry 3 }	INTEGER none(1) on(2) off(3)	read-only	current	N
rs232OutSigChanges	{ rs232OutSigEntry 4 }	Counter32	read-only	current	N

5.10.3.6.1 rs232OutSigPortIndex

The value of rs232PortIndex, Section 5.10.3.2.1, Page 162, for the port to which this entry belongs.

5.10.3.6.2 rs232OutSigName

Identification of a hardware signal, as follows:

- rts - Request to Send
- cts - Clear to Send
- dsr - Data Set Ready
- dtr - Data Terminal Ready
- ri - Ring Indicator
- dcd - Received Line Signal Detector
- sq - Signal Quality Detector
- srs - Data Signaling Rate Selector
- srts - Secondary Request to Send
- scts - Secondary Clear to Send
- sdcd - Secondary Received Line Signal Detector

5.10.3.6.3 rs232OutSigState

The current signal state.

- none(1)
- on(2)
- off(3)

5.10.3.6.4 rs232OutSigChanges

The number of times the signal has changed from “on” to “off” or from “off” to “on.”

5.11 ADSL-LINE-MIB — RFC 2662

5.11.1 Introduction

This MIB defines objects for the management of ADSL lines for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. The RFC used as a reference was dated August 1999 and the ADSL Line MIB has a LAST-UPDATED of "9908190000Z."

Not all objects defined in the MIB are supported by the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. What is supported or not supported is identified.

In addition, memory required for the various MIB objects is identified.

5.11.1.1 Textual Conventions

The following textual conventions are made in the RFC:

- AdslLineCodingType - This data type is used as the syntax for the ADSL Line Code.
  - SYNTAX INTEGER
    - other(1) - none of the following
    - dmt(2) - Discrete MultiTone
    - cap(3) - Carrierless Amplitude & Phase modulation
    - qam(4) - Quadrature Amplitude Modulation
- AdslPerfCurrDayCount - A counter associated with interface performance measurements in a current 1-day (24 hour) measurement interval.

  The value of this counter starts at zero at the beginning of an interval and is increased when a associated events occur, until the end of the 1-day interval. At that time the value of the counter is stored in the previous 1-day history interval, if available, and the current interval counter is restarted at zero.

  In the case where the agent has no valid data available for this interval the corresponding object instance is not available and upon a retrieval request a corresponding error message shall be returned to indicate that this instance does not exist (for example, a noSuchName error for SNMPv1 and a noSuchInstance for SNMPv2 GET operation).

  - SYNTAX Gauge32
- AdslPerfPrevDayCount - A counter associated with interface performance measurements during the most previous 1-day (24 hour) measurement interval. The value of this counter is equal to the value of the current day counter at the end of its most recent interval.

  In the case where the agent has no valid data available for this interval the corresponding object instance is not available and upon a retrieval request a corresponding error message shall be returned to indicate that this instance does not exist (for example, a noSuchName error for SNMPv1 and a noSuchInstance for SNMPv2 GET operation).

  - SYNTAX Gauge32
- AdslPerfTimeElapsed - The number of seconds that have elapsed since the beginning of the current measurement period. If, for some reason, such as an adjustment in the system's time-of-day clock, the current interval exceeds the maximum value, the agent will return the maximum value.
  - SYNTAX Gauge32

5.11.1.2 Naming Conventions

- Atuc - (ATUC) modem at near (Central) end of line

- Atur - (ATUR) modem at Remote end of line
- Curr - Current
- Prev - Previous
- Atn - Attenuation
- ES - Errored Second.
- LCS - Line Code Specific
- Lof - Loss of Frame
- Lol - Loss of Link
- Los - Loss of Signal
- Lpr - Loss of Power
- Ixxxs - interval of Seconds in which xxx occurs (for example, xxx=Lof, Los, Lpr)
- Max - Maximum
- Mgn - Margin
- Min - Minimum
- Psd - Power Spectral Density
- Snr - Signal to Noise Ratio
- Tx - Transmit
- Blks - Blocks, a data unit, see adslAtuXChanCrcBlockLength

5.11.1.3 OIDs

• iso	- { 1 }	1
• org	- { iso 3 }	1.3
• dod	- { org 6 }	1.3.6
• internet	- { dod 1 }	1.3.6.1
• mgmt	- { internet 2 }	1.3.6.1.2
• mib-2	- { mgmt 1 }	1.3.6.1.2.1
• transmission	- { mib-2 10 }	1.3.6.1.2.1.10
• adslMIB	- { transmission 94 }	1.3.6.1.2.1.10.94
• adslLineMib	- { adslMIB 1 }	1.3.6.1.2.1.10.94.1
• adslMibObjects	- { adslLineMib 1 }	1.3.6.1.2.1.10.94.1.1
• adslTraps	- { adslLineMib 2 }	1.3.6.1.2.1.10.94.1.2
• adsAtucTraps	- { adslTraps 1 }	1.3.6.1.2.1.10.94.1.2.1
• adsAturTraps	- { adslTraps 2 }	1.3.6.1.2.1.10.94.1.2.2
• adslLCSMib	- { adslMibObjects 16 }	1.3.6.1.2.1.10.94.1.1.16

5.11.2 MIB Objects

This MIB has multiple parallel tables.

- line - common attributes
- atuc and atur status
- atuc and atur performance
  - — Current and up to 96 buckets of 15 min performance history
  - — Current and Previous 1-day bucket performance history
- profiles

Table 95. ADSL-LINE-MIB Objects

Object	OID	Syntax SEQUENCE of	Access	Status	Supported
adslLineTable	{ adslMibObjects 1 }	adslLineEntry	not-accessible	current	Y
adslAtucPhysTable	{ adslMibObjects 2 }	adslAtucPhysEntry	not-accessible	current	Y
adslAturPhysTable	{ adslMibObjects 3 }	adslAturPhysEntry	not-accessible	current	Y
adslAtucChanTable	{ adslMibObjects 4 }	adslAtucChanEntry	not-accessible	current	Y
adslAturChanTable	{ adslMibObjects 5 }	adslAturChanEntry	not-accessible	current	Y
adslAtucPerfDataTable	{ adslMibObjects 6 }	adslAtucPerfDataEntry	not-accessible	current	Y
adslAturPerfDataTable	{ adslMibObjects 7 }	adslAturPerfDataEntry	not-accessible	current	Y
adslAtucIntervalTable	{ adslMibObjects 8 }	adslAtucIntervalEntry	not-accessible	current	Y
adslAturIntervalTable	{ adslMibObjects 9 }	adslAturIntervalEntry	not-accessible	current	Y
adslAtucChanPerfDataTable	{ adslMibObjects 10 }	adslAtucChanPerfDataEntry	not-accessible	current	Y
adslAturChanPerfDataTable	{ adslMibObjects 11 }	adslAturChanPerfDataEntry	not-accessible	current	Y
adslAtucChanIntervalTable	{ adslMibObjects 12 }	adslAtucChanIntervalEntry	not-accessible	current	Y
adslAturChanIntervalTable	{ adslMibObjects 13 }	adslAturChanIntervalEntry	not-accessible	current	Y
adslLineConfProfileTable	{ adslMibObjects 14 }	adslLineConfProfileEntry	not-accessible	current	Y
adslLineAlarmConfProfileTable	{ adslMibObjects 15 }	adslLineAlarmConfProfileEntry	not-accessible	current	Y

5.11.2.1 adslLineTable

The ADSL Line Table includes common attributes describing both ends of the line. It is required for all ADSL physical interfaces. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

This table is a sequence of adslLineEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslLineTable - { adslMibObjects 1 } 1.3.6.1.2.1.10.94.1.1.1
- adslLineEntry - { adslLineTable 1} 1.3.6.1.2.1.10.94.1.1.1.2

Table 96. adslLineTable

Object	OID	Syntax	Access	Status	Supported
adslLineCoding	{ adslLineEntry 1 }	AdslLineCodingType	read-only	current	Y

Table 96. adslLineTable

Object	OID	Syntax	Access	Status	Supported
adslLineType	{ adslLineEntry 2 }	INTEGER noChannel(1) fastOnly(2) interleavedOnly(3) fastOrInterleaved(4) fastAndInterLeaved(5)	read-only	current	Y
adslLineSpecific	{ adslLineEntry 3 }	VariablePointer	read-only	current	Y
adslLineConfProfile	{ adslLineEntry 4 }	SnmpAdminString (SIZE (1…32))	read-write	current	Y
adslLineAlarmConfProfile	{ adslLineEntry 5 }	SnmpAdminString (SIZE (1…32))	read-write	current	Y

5.11.2.1.1 adslLineCoding

Specifies the ADSL coding type used on this line.

Valid Object Value(s)

- ADSL
  - dmt(2)
- reachDSL
  - other(1)

5.11.2.1.2 adslLineType

Defines the type of ADSL physical line entity that exists, by defining whether and how the line is channelized. If the line is channelized, the value will be other than noChannel(1). This object defines which channel type(s) are supported.

In the case that the line is channelized, the manager can use the ifStackTable to determine the ifIndex for the associated channel(s).

- noChannel (1) - no channels exist
- fastOnly (2) - fast channel exists only
- interleavedOnly (3) - interleaved channel exists only
- fastOrInterleaved (4) - either fast or interleaved channels can exist, but only one at any time
- fastAndInterleaved (5) - both fast or interleaved channels exist

Valid Object Value(s)

- ADSL
  - fastOrInterleaved (4)
- reachDSL
  - fastOnly(2)

5.11.2.1.3 adslLineSpecific

OID instance in vendor-specific MIB. The Instance may be used to determine shelf/slot/port of the ATUC interface in a DSLAM.

5.11.2.1.4 adslLineConfProfile

The value of this object identifies the row in the ADSL Line Configuration Profile Table, (adslLineConfProfileTable, Section 5.11.2.14, Page 198) which applies for this ADSL Line, and channels if applicable.

For “dynamic” mode, in the case which the configuration profile has not been set, the value is set to “DEFVAL.” If the implementator of this MIB has chosen not to implement “dynamic assignment” of profiles, this object's MIN-ACCESS is read-only.

5.11.2.1.5 adslLineAlarmConfProfile

The value of this object identifies the row in the ADSL Line Alarm Configuration Profile Table, adslLineAlarmConfProfileTable, Section 5.11.2.15, Page 204) which applies to this ADSL line, and channels if applicable.

For “dynamic” mode, in the case which the alarm profile has not been set, the value is set to “DEFVAL.”

If the implementator of this MIB has chosen not to implement “dynamic assignment” of profiles, this object's MIN-ACCESS is read-only.

5.11.2.2 adslAtucPhysTable

The ADSL ATU-C Physical Table provides one row for each ATUC. Each row contains the Physical Layer Parameters table for that ATUC. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

This table is a sequence of adslAtucPhysEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAtucPhysTable - { adslMibObjects 2 } 1.3.6.1.2.1.10.94.1.1.2
- adslAtucPhysEntry - { adslAtucPhysTable 1} 1.3.6.1.2.1.10.94.1.1.2.1

Table 97. adslAtucPhysTable

Object	OID	Syntax	Access	Status	Supported
adslAtucInvSerialNumber	{ adslAtucPhysEntry 1 }	SnmpAdminString (SIZE (0…32))	read-only	current	Y
adslAtucInvVendorID	{ adslAtucPhysEntry 2 }	SnmpAdminString (SIZE (0…16))	read-only	current	Y
adslAtucInvVersionNumber	{ adslAtucPhysEntry 3 }	SnmpAdminString (SIZE (0…16))	read-only	current	Y
adslAtucCurrSnrMgn	{ adslAtucPhysEntry 4 }	INTEGER (-640…640)	read-only	current	Y
adslAtucCurrAtn	{ adslAtucPhysEntry 5 }	Gauge32(0…630)	read-only	current	Y
adslAtucCurrStatus	{ adslAtucPhysEntry 6 }	BITS	read-only	current	Y
adslAtucCurrOutputPwr	{ adslAtucPhysEntry 7 }	INTEGER	read-only	current	Y
adslAtucCurrAttainableRate	{ adslAtucPhysEntry 8 }	Gauge32	read-only	current	Y

5.11.2.2.1 adslAtucInvSerialNumber

The vendor specific string that identifies the vendor equipment.

Valid Object Value(s)

The value of this object is the serial number of chassis the port is associated with.

5.11.2.2.2 adslAtucInvVendorID

The vendor ID code is a copy of the binary vendor identification field defined by the ANSI T1.413-1995 specification and expressed as readable characters.

Valid Object Value(s)

This object will take on a value of 0011b as assigned to “Paradyne” in Annex D of Issue 2 of the T1.413 specification.

5.11.2.2.3 adslAtucInvVersionNumber

The vendor specific version number sent by this ATU as part of the initialization messages. It is a copy of the binary version number field defined by the ANSI T1.413-1995 specification and expressed as readable characters.

Valid Object Value(s)

The value of this object will indicate a version number that is significant for upgrades, diagnostics, maintenance, etc.

5.11.2.2.4 adslAtucCurrSnrMgn

Noise Margin as seen by this ATU with respect to its received signal in tenths of a decibel.

5.11.2.2.5 adslAtucCurrAtn

Measured difference in the total power transmitted by the peer ATU and the total power received by this ATU in tenths of a decibel.

5.11.2.2.6 adslAtucCurrStatus

Indicates current state of the ATUC line. This is a bit-map of possible conditions. The various bit positions are:

- noDefect(0) - There no defects on the line
- lossOfFraming(1) - ATUC failure due to not receiving valid frame
- lossOfSignal(2) - ATUC failure due to not receiving signal
- lossOfPower(3) - ATUC failure due to loss of power
- lossOfSignalQuality(4) - Loss of Signal Quality is declared when the Noise Margin falls below the Minimum Noise Margin, or the bit-error-rate exceeds 10^-7
- lossOfLink(5) - TUC failure due to inability to link with ATUR
- dataInitFailure(6) - ATUC failure during initialization due to bit errors corrupting start-up exchange data
- configInitFailure(7) - ATUC failure during initialization due to peer ATU not able to support requested configuration
- protocolInitFailure(8) - ATUC failure during initialization due to incompatible protocol used by the peer ATU
- noPeerAtuPresent(9) - ATUC failure during initialization due to no activation sequence detected from peer ATU

This is intended to supplement ifOperStatus.

5.11.2.2.7 adslAtucCurrOutputPwr

Measured total output power, in tenths of a dBm, transmitted by this ATU. This is the measurement that was reported during the last activation sequence.

5.11.2.2.8 adslAtucCurrAttainableRate

Indicates the maximum currently attainable data rate by the ATU. This value is equal or greater than the current line rate.

5.11.2.3 adslAturPhysTable

This table provides one row for each ATUR Each row contains the Physical Layer Parameters table for that ATUR. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

This table is a sequence of adslAturPhysEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAturPhysTable - { adslMibObjects 3 } 1.3.6.1.2.1.10.94.1.1.3
- adslAturPhysEntry - { adslAturPhysTable 1} 1.3.6.1.2.1.10.94.1.1.3.1

Table 98. adslAturPhysTable

Object	OID	Syntax	Access	Status	Supported
adslAturInvSerialNumber	{ adslAturPhysEntry 1 }	SnmpAdminString (SIZE (0…32))	read-only	current	Y
adslAturInvVendorID	{ adslAturPhysEntry 2 }	SnmpAdminString (SIZE (0…16))	read-only	current	Y
adslAturInvVersionNumber	{ adslAturPhysEntry 3 }	SnmpAdminString (SIZE (0…16))	read-only	current	Y
adslAturCurrSnrMgn	{ adslAturPhysEntry 4 }	INTEGER (-640…640)	read-only	current	Y
adslAturCurrAtn	{ adslAturPhysEntry 5 }	Gauge32 (0…630)	read-only	current	Y
adslAturCurrStatus	{ adslAturPhysEntry 6 }	BITS	read-only	current	Y
adslAturCurrOutputPwr	{ adslAturPhysEntry 7 }	INTEGER	read-only	current	Y
adslAturCurrAttainableRate	{ adslAturPhysEntry 8 }	Gauge32	read-only	current	Y

5.11.2.3.1 adslAturInvSerialNumber

The vendor specific string that identifies the vendor equipment.

5.11.2.3.2 adslAturInvVendorID

The vendor ID code is a copy of the binary vendor identification field defined by the ANSI T1.413-1995 specification and expressed as readable characters.

5.11.2.3.3 adslAturInvVersionNumber

The vendor specific version number sent by this ATU as part of the initialization messages. It is a copy of the binary version number field defined by the ANSI T1.413-1995 specification and expressed as readable characters.

5.11.2.3.4 adslAturCurrSnrMgn

Noise Margin as seen by this ATU with respect to its received signal in tenths of a decibel.

5.11.2.3.5 adslAturCurrAtn

Measured difference in the total power, in tenths of a decibel, transmitted by the peer ATU and the total power received by this ATU.

5.11.2.3.6 adslAturCurrStatus

Indicates current state of the ATUR line. This is a bit-map of possible conditions. Due to the isolation of the ATUR when line problems occur, many state conditions like loss of power, loss of quality signal, and initialization errors, can not be determined. While trouble shooting ATUR, also use object, adslAtucCurrStatus. The various bit positions are:

- noDefect(0) - There no defects on the line
- lossOfFraming(1) - ATUR failure due to not receiving valid frame
- lossOfSignal(2) - ATUR failure due to not receiving signal
- lossOfPower(3) - ATUR failure due to loss of power
- lossOfSignalQuality(4) - Loss of Signal Quality is declared when the Noise Margin falls below the Minimum Noise Margin, or the bit-error-rate exceeds 10^-7

This is intended to supplement ifOperStatus.

5.11.2.3.7 adslAturCurrOutputPwr

Measured total output power transmitted by this ATU. This is the measurement that was reported during the last activation sequence.

5.11.2.3.8 adslAturCurrAttainableRate

Indicates the maximum currently attainable data rate by the ATU. This value is equal or greater than the current line rate.

5.11.2.4 adslAtucChanTable

This table provides one row for each ATUC channel. ADSL channel interfaces are those ifEntries where ifType is equal to adslInterleave(124) or adslFast(125).

This table is a sequence of adslAtucChanEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAtucChanTable - { adslMibObjects 4 } 1.3.6.1.2.1.10.94.1.1.4
- adslAtucChanEntry - { adslAtucChanTable 1} 1.3.6.1.2.1.10.94.1.1.4.1

Table 99. adslAutcChanTable

Object	OID	Syntax	Access	Status	Supported
adslAtucChanInterleaveDelay	{ adslAtucChanEntry 1 }	Gauge32	read-only	current	Y
adslAtucChanCurrTxRate	{ adslAtucChanEntry 2 }	Gauge32	read-only	current	Y
adslAtucChanPrevTxRate	{ adslAtucChanEntry 3 }	Gauge32	read-only	current	Y
adslAtucChanCrcBlockLength	{ adslAtucChanEntry 4 }	Gauge32	read-only	current	Y

5.11.2.4.1 adslAtucChanInterleaveDelay

Interleave Delay for this channel in milli-seconds.

Interleave delay applies only to the interleave channel and defines the mapping (relative spacing) between subsequent input bytes at the interleaver input and their placement in the bit stream at the interleaver output. Larger numbers

provide greater separation between consecutive input bytes in the output bit stream allowing for improved impulse noise immunity at the expense of payload latency.

In the case where the ifType is Fast(125), use noSuchObject.

5.11.2.4.2 adslAtucChanCurrTxRate

Actual transmit rate on this channel in bps.

5.11.2.4.3 adslAtucChanPrevTxRate

The rate, in bps, at the time of the last adslAtucRateChangeTrap event. It is also set at initialization to prevent a trap being sent.

Rate changes less than adslAtucThresh (*) RateDown or less than adslAtucThresh (*) RateUp will not cause a trap or cause this object to change.

(*) == Fast or Interleave.

See adslLineAlarmConfProfileTable, Section 5.11.2.15, Page 204.

5.11.2.4.4 adslAtucChanCrcBlockLength

Indicates the length, in bytes, of the channel data-block on which the CRC operates. Reference the following Line Code Specific MIBs for more information:

- ADSL Forum WT-014, “DMT Line Code Specific MIB,” February 1999
- ADSL Forum WT-015, “CAP Line Code Specific MIB,” February 1999

5.11.2.5 adslAturChanTable

This table provides one row for each ATUR channel. ADSL channel interfaces are those ifEntries where ifType is equal to adslInterleave(124) or adslFast(125).

This table is a sequence of adslAturChanEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAturChanTable - { adslMibObjects 5 } 1.3.6.1.2.1.10.94.1.1.5
- adslAturChanEntry - { adslAturChanTable 1 } 1.3.6.1.2.1.10.94.1.1.5.1

Table 100. adslAturChanTable

Object	OID	Syntax	Access	Status	Supported
adslAturChanInterleaveDelay	{ adslAturChanEntry 1 }	Gauge32	read-only	current	Y
adslAturChanCurrTxRate	{ adslAturChanEntry 2 }	Gauge32	read-only	current	Y
adslAturChanPrevTxRate	{ adslAturChanEntry 3 }	Gauge32	read-only	current	Y
adslAturChanCrcBlockLength	{ adslAturChanEntry 4 }	Gauge32	read-only	current	Y

5.11.2.5.1 adslAturChanInterleaveDelay

Interleave Delay for this channel in units of milli-seconds.

Interleave delay applies only to the interleave channel and defines the mapping (relative spacing) between subsequent input bytes at the interleaver input and their placement in the bit stream at the interleaver output. Larger numbers

provide greater separation between consecutive input bytes in the output bit stream allowing for improved impulse noise immunity at the expense of payload latency.

In the case where the ifType is Fast(125), use noSuchObject.

5.11.2.5.2 adslAturChanCurrTxRate

The actual transmit rate on this channel. in bps.

5.11.2.5.3 adslAturChanPrevTxRate

The rate, in bps, at the time of the last adslAturRateChangeTrap event. It is also set at initialization to prevent a trap being sent. Rate changes less than adslAturThresh (*) RateDown or less than adslAturThresh (*) RateUp will not cause a trap or cause this object to change.

(*) == Fast or Interleave.

See adslLineAlarmConfProfileTable, Section 5.11.2.15, Page 204.

5.11.2.5.4 adslAturChanCrcBlockLength

Indicates the length of the channel data-block on which the CRC operates. Reference the following Line Code Specific MIBs for more information:

- ADSL Forum WT-014, “DMT Line Code Specific MIB,” February 1999
- ADSL Forum WT-015, “CAP Line Code Specific MIB,” February 1999

5.11.2.6 adslAtucPerfDataTable

This table provides one row for each ATUC. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

This table is a sequence of adslAtucPerfDataEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAtucPerfDataTable - { adslMibObjects 6 } 1.3.6.1.2.1.10.94.1.1.6
- adslAtucPerfDataEntry - { adslAtucPerfDataTable 1 } 1.3.6.1.2.1.10.94.1.1.6.1

Table 101. adslAtucPerfDataTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
adslAtucPerfLofs	{ adslAtucPerfDataEntry 1 }	Counter32	read-only	current	Y
adslAtucPerfLoss	{ adslAtucPerfDataEntry 2 }	Counter32	read-only	current	Y
adslAtucPerfLols	{ adslAtucPerfDataEntry 3 }	Counter32	read-only	current	Y
adslAtucPerfLprs	{ adslAtucPerfDataEntry 4 }	Counter32	read-only	current	N
adslAtucPerfESs	{ adslAtucPerfDataEntry 5 }	Counter32	read-only	current	Y
adslAtucPerfInits	{ adslAtucPerfDataEntry 6 }	Counter32	read-only	current	Y
adslAtucPerfValidIntervals	{ adslAtucPerfDataEntry 7 }	INTEGER (0…96)	read-only	current	Y
adslAtucPerfInvalidIntervals	{ adslAtucPerfDataEntry 8 }	INTEGER (0…96)	read-only	current	Y

Table 101. adslAtucPerfDataTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
adslAtucPerfCurr15MinTimeElapsed	{ adslAtucPerfDataEntry 9 }	AdslPerfTimeElapsed (0…899)	read-only	current	Y
adslAtucPerfCurr15MinLofs	{ adslAtucPerfDataEntry 10 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinLoss	{ adslAtucPerfDataEntry 11 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinLols	{ adslAtucPerfDataEntry 12 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinLprs	{ adslAtucPerfDataEntry 13 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinESs	{ adslAtucPerfDataEntry 14 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinInits	{ adslAtucPerfDataEntry 15 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr1DayTimeElapsed	{ adslAtucPerfDataEntry 16 }	AdslPerfTimeElapsed	read-only	current	Y
adslAtucPerfCurr1DayLofs	{ adslAtucPerfDataEntry 17 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayLoss	{ adslAtucPerfDataEntry 18 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayLols	{ adslAtucPerfDataEntry 19 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayLprs	{ adslAtucPerfDataEntry 20 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayESs	{ adslAtucPerfDataEntry 21 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayInits	{ adslAtucPerfDataEntry 22 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfPrev1DayMoniSecs	{ adslAtucPerfDataEntry 23 }	INTEGER(0…86400)	read-only	current	Y
adslAtucPerfPrev1DayLofs	{ adslAtucPerfDataEntry 24 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DayLoss	{ adslAtucPerfDataEntry 25 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DayLols	{ adslAtucPerfDataEntry 26 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DayLprs	{ adslAtucPerfDataEntry 27 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DayESs	{ adslAtucPerfDataEntry 28 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DayInits	{ adslAtucPerfDataEntry 29 }	AdslPerfPrevDayCount	read-only	current	Y

Also see adslAtucIntervalTable for 15 minute interval elapsed counters.

5.11.2.6.1 adslAtucPerfLofs

Count of the number of Loss of Framing failures since agent reset.

5.11.2.6.2 adslAtucPerfLoss

Count of the number of Loss of Signal failures since agent reset.

5.11.2.6.3 adslAtucPerfLols

Count of the number of Loss of Link failures since agent reset.

5.11.2.6.4 adslAtucPerfLprs

Count of the number of Loss of Power failures since agent reset.

5.11.2.6.5 adslAtucPerfESs

Count of the number of Errored Seconds since agent reset. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.6.6 adslAtucPerfInits

Count of the line initialization attempts since agent reset. Includes both successful and failed attempts.

5.11.2.6.7 adslAtucPerfValidIntervals

The number of previous 15-minute intervals in the interval table for which data was collected. Given that <n> is the maximum # of intervals supported. The value is <n> unless the measurement was (re-)started within the last (<n>*15) minutes, in which case the value is the number of complete 15 minute intervals for which the agent has at least some data. In certain cases (for example, in the case where the agent is a proxy) it is possible that some intervals are unavailable. In this case, this interval is the maximum interval number for which data is available.

5.11.2.6.8 adslAtucPerfInvalidIntervals

The number of intervals in the range from 0 to the value of adslAtucPerfValidIntervals for which no data is available. This object will typically be zero except in cases where the data for some intervals are not available (for example, in proxy situations).

5.11.2.6.9 adslAtucPerfCurr15MinTimeElapsed

Total elapsed seconds in this interval.

5.11.2.6.10 adslAtucPerfCurr15MinLofs

Count of seconds in the current 15 minute interval when there was Loss of Framing.

5.11.2.6.11 adslAtucPerfCurr15MinLoss

Count of seconds in the current 15 minute interval when there was Loss of Signal.

5.11.2.6.12 adslAtucPerfCurr15MinLols

Count of seconds in the current 15 minute interval when there was Loss of Link.

5.11.2.6.13 adslAtucPerfCurr15MinLprs

Count of seconds in the current 15 minute interval when there was Loss of Power.

5.11.2.6.14 adslAtucPerfCurr15MinESs

Count of Errored Seconds in the current 15 minute interval. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.6.15 adslAtucPerfCurr15MinInits

Count of the line initialization attempts in the current 15 minute interval. Includes both successful and failed attempts.

5.11.2.6.16 adslAtucPerfCurr1DayTimeElapsed

Number of seconds that have elapsed since the beginning of the current 1-day interval.

5.11.2.6.17 adslAtucPerfCurr1DayLofs

Count of the number of seconds when there was Loss of Framing during the current day as measured by adslAtucPerfCurr1DayTimeElapsed.

5.11.2.6.18 adslAtucPerfCurr1DayLoss

Count of the number of seconds when there was Loss of Signal during the current day as measured by adslAtucPerfCurr1DayTimeElapsed.

5.11.2.6.19 adslAtucPerfCurr1DayLols

Count of the number of seconds when there was Loss of Link during the current day as measured by adslAtucPerfCurr1DayTimeElapsed.

5.11.2.6.20 adslAtucPerfCurr1DayLprs

Count of the number of seconds when there was Loss of Power during the current day as measured by adslAtucPerfCurr1DayTimeElapsed.

5.11.2.6.21 adslAtucPerfCurr1DayESs

Count of Errored Seconds during the current day as measured by adslAtucPerfCurr1DayTimeElapsed. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.6.22 adslAtucPerfCurr1DayInits

Count of the line initialization attempts in the day as measured by adslAtucPerfCurr1DayTimeElapsed. Includes both successful and failed attempts.

5.11.2.6.23 adslAtucPerfPrev1DayMoniSecs

The amount of time in the previous 1-day interval over which the performance monitoring information is actually counted. This value is the same as the interval duration except in a situation where performance monitoring data could not be collected for any reason.

5.11.2.6.24 adslAtucPerfPrev1DayLofs

Count of seconds in the interval when there was Loss of Framing within the most recent previous 1-day period.

5.11.2.6.25 adslAtucPerfPrev1DayLoss

Count of seconds in the interval when there was Loss of Signal within the most recent previous 1-day period.

5.11.2.6.26 adslAtucPerfPrev1DayLols

Count of seconds in the interval when there was Loss of Link within the most recent previous 1-day period.

5.11.2.6.27 adslAtucPerfPrev1DayLprs

Count of seconds in the interval when there was Loss of Power within the most recent previous 1-day period.

5.11.2.6.28 adslAtucPerfPrev1DayESs

Count of Errored Seconds within the most recent previous 1-day period. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.6.29 adslAtucPerfPrev1DayInits

Count of the line initialization attempts in the most recent previous 1-day period. Includes both successful and failed attempts.

5.11.2.7 adslAturPerfDataTable

This table provides one row for each ATUR. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

This table is a sequence of adslAturPerfDataEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAturPerfDataTable - { adslMibObjects 7 } 1.3.6.1.2.1.10.94.1.1.7

- adslAturPerfDataEntry - { adslAturPerfDataTable 1 } 1.3.6.1.2.1.10.94.1.1.7.1

Table 102. adslAturPerfDataTable

Object	OID	Syntax	Access	Status	Supported
adslAturPerfLofs	{ adslAturPerfDataEntry 1 }	Counter32	read-only	current	Y
adslAturPerfLoss	{ adslAturPerfDataEntry 2 }	Counter32	read-only	current	Y
adslAturPerfLprs	{ adslAturPerfDataEntry 3 }	Counter32	read-only	current	Y
adslAturPerfESs	{ adslAturPerfDataEntry 4 }	Counter32	read-only	current	Y
adslAturPerfValidIntervals	{ adslAturPerfDataEntry 5 }	INTEGER (0…96)	read-only	current	Y
adslAturPerfInvalidIntervals	{ adslAturPerfDataEntry 6 }	INTEGER (0…96)	read-only	current	Y
adslAturPerfCurr15MinTimeElapsed	{ adslAturPerfDataEntry 7 }	AdslPerfTimeElapsed (0…899)	read-only	current	Y
adslAturPerfCurr15MinLofs	{ adslAturPerfDataEntry 8 }	PerfCurrentCount	read-only	current	Y
adslAturPerfCurr15MinLoss	{ adslAturPerfDataEntry 9 }	PerfCurrentCount	read-only	current	Y
adslAturPerfCurr15MinLprs	{ adslAturPerfDataEntry 10 }	PerfCurrentCount	read-only	current	Y
adslAturPerfCurr15MinESs	{ adslAturPerfDataEntry 11 }	PerfCurrentCount	read-only	current	Y
adslAturPerfCurr1DayTimeElapsed	{ adslAturPerfDataEntry 12 }	AdslPerfTimeElapsed	read-only	current	Y
adslAturPerfCurr1DayLofs	{ adslAturPerfDataEntry 13 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturPerfCurr1DayLoss	{ adslAturPerfDataEntry 14 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturPerfCurr1DayLprs	{ adslAturPerfDataEntry 15 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturPerfCurr1DayESs	{ adslAturPerfDataEntry 16 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturPerfPrev1DayMoniSecs	{ adslAturPerfDataEntry 17 }	INTEGER (0…86400)	read-only	current	Y
adslAturPerfPrev1DayLofs	{ adslAturPerfDataEntry 18 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturPerfPrev1DayLoss	{ adslAturPerfDataEntry 19 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturPerfPrev1DayLprs	{ adslAturPerfDataEntry 20 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturPerfPrev1DayESs	{ adslAturPerfDataEntry 21 }	AdslPerfPrevDayCount	read-only	current	Y

5.11.2.7.1 adslAturPerfLofs

Count of the number of Loss of Framing failures since agent reset.

5.11.2.7.2 adslAturPerfLoss

Count of the number of Loss of Signal failures since agent reset.

5.11.2.7.3 adslAturPerfLprs

Count of the number of Loss of Power failures since agent reset.

5.11.2.7.4 adslAturPerfESs

Count of the number of Errored Seconds since agent reset. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.7.5 adslAturPerfValidIntervals

The number of previous 15-minute intervals in the interval table for which data was collected. Given that <n> is the maximum # of intervals supported. The value is <n> unless the measurement was (re-)started within the last (<n>*15) minutes, in which case the value is the number of complete 15 minute intervals for which the agent has at least some data. In certain cases (for example, in the case where the agent is a proxy) it is possible that some intervals are unavailable. In this case, this interval is the maximum interval number for which data is available.

5.11.2.7.6 adslAturPerfInvalidIntervals

The number of intervals in the range from 0 to the value of adslAturPerfValidIntervals for which no data is available. This object will typically be zero except in cases where the data for some intervals are not available (for example, in proxy situations).

5.11.2.7.7 adslAturPerfCurr15MinTimeElapsed

Total elapsed seconds in this interval.

5.11.2.7.8 adslAturPerfCurr15MinLofs

Count of seconds in the current 15 minute interval when there was Loss of Framing.

5.11.2.7.9 adslAturPerfCurr15MinLoss

Count of seconds in the current 15 minute interval when there was Loss of Signal.

5.11.2.7.10 adslAturPerfCurr15MinLprs

Count of seconds in the current 15 minute interval when there was Loss of Power.

5.11.2.7.11 adslAturPerfCurr15MinESs

Count of Errored Seconds in the current 15 minute interval. The errored second parameter is a count of one-second intervals containing one or more crc nomalies, or one or more los or sef defects.

5.11.2.7.12 adslAturPerfCurr1DayTimeElapsed

Number of seconds that have elapsed since the beginning of the current 1-day interval.

5.11.2.7.13 adslAturPerfCurr1DayLofs

Count of the number of seconds when there was Loss of Framing during the current day as measured by adslAturPerfCurr1DayTimeElapsed.

5.11.2.7.14 adslAturPerfCurr1DayLoss

Count of the number of seconds when there was Loss of Signal during the current day as measured by adslAturPerfCurr1DayTimeElapsed.

5.11.2.7.15 adslAturPerfCurr1DayLprs

Count of the number of seconds when there was Loss of Power during the current day as measured by adslAturPerfCurr1DayTimeElapsed.

5.11.2.7.16 adslAturPerfCurr1DayESs

Count of Errored Seconds during the current day as measured by adslAturPerfCurr1DayTimeElapsed. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.7.17 adslAturPerfPrev1DayMoniSecs

The amount of time in the previous 1-day interval over which the performance monitoring information is actually counted. This value is the same as the interval duration except in a situation where performance monitoring data could not be collected for any reason.

5.11.2.7.18 adslAturPerfPrev1DayLofs

Count of seconds in the interval when there was Loss of Framing within the most recent previous 1-day period.

5.11.2.7.19 adslAturPerfPrev1DayLoss

Count of seconds in the interval when there was Loss of Signal within the most recent previous 1-day period.

5.11.2.7.20 adslAturPerfPrev1DayLprs

Count of seconds in the interval when there was Loss of Power within the most recent previous 1-day period.

5.11.2.7.21 adslAturPerfPrev1DayESs

Count of Errored Seconds within the most recent previous 1-day period. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.8 adslAtucIntervalTable

This table provides one row for each ATUC performance data collection interval. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

This table is a sequence of adslAtucIntervalEntry.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- adslAtucIntervalNumber

OIDs:

- adslAtucIntervalTable - { adslMibObjects 8 } 1.3.6.1.2.1.10.94.1.1.8
- adslAtucIntervalEntry - { adslAtucIntervalTable 1 } 1.3.6.1.2.1.10.94.1.1.8.1

Table 103. adslAtucIntervalTable

Object	OID	Syntax	Access	Status	Supported
adslAtucIntervalNumber	{ adslAtucIntervalEntry 1 }	INTEGER(1…96)	read-only	current	Y
adslAtucIntervalLofs	{ adslAtucIntervalEntry 2 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalLoss	{ adslAtucIntervalEntry 3 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalLols	{ adslAtucIntervalEntry 4 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalLprs	{ adslAtucIntervalEntry 5 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalESs	{ adslAtucIntervalEntry 6 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalInits	{ adslAtucIntervalEntry 7 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalValidData	{ adslAtucIntervalEntry 8 }	TruthValue	read-only	current	Y

5.11.2.8.1 adslAtucIntervalNumber

Performance Data Interval number 1 is the most recent previous interval; interval 96 is 24 hours ago. Intervals 2…96 are optional.

5.11.2.8.2 adslAtucIntervalLofs

Count of seconds in the interval when there was Loss of Framing.

5.11.2.8.3 adslAtucIntervalLoss

Count of seconds in the interval when there was Loss of Signal.

5.11.2.8.4 adslAtucIntervalLols

Count of seconds in the interval when there was Loss of Link.

5.11.2.8.5 adslAtucIntervalLprs

Count of seconds in the interval when there was Loss of Power.

5.11.2.8.6 adslAtucIntervalESs

Count of Errored Seconds in the interval. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.8.7 adslAtucIntervalInits

Count of the line initialization attempts during the interval. Includes both successful and failed attempts.

5.11.2.8.8 adslAtucIntervalValidData

This variable indicates if the data for this interval is valid.

5.11.2.9 adslAturIntervalTable

This table provides one row for each ATUR performance data collection interval. ADSL physical interfaces are those ifEntries where ifType is equal to adsl(94).

The index for this table is the ifIndex and adslAturIntervalNumber objects. This table is a sequence of adslAturIntervalEntry.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- adslAturIntervalNumber

OIDs:

- adslAturIntervalTable - { adslMibObjects 9 } 1.3.6.1.2.1.10.94.1.1.9
- adslAturIntervalEntry - { adslAturIntervalTable 1 } 1.3.6.1.2.1.10.94.1.1.9.1

Table 104. adslAturIntervalTable

Object	OID	Syntax	Access	Status	Supported
adslAturIntervalNumber	{ adslAturIntervalEntry 1 }	INTEGER(1…96)	read-only	current	Y
adslAturIntervalLofs	{ adslAturIntervalEntry 2 }	PerfIntervalCount	read-only	current	Y
adslAturIntervalLoss	{ adslAturIntervalEntry 3 }	PerfIntervalCount	read-only	current	Y
adslAturIntervalLprs	{ adslAturIntervalEntry 4 }	PerfIntervalCount	read-only	current	Y
adslAturIntervalESs	{ adslAturIntervalEntry 5 }	PerfIntervalCount	read-only	current	Y

Table 104. adslAturIntervalTable

Object	OID	Syntax	Access	Status	Supported
adslAturIntervalValidData	{ adslAturIntervalEntry 6 }	TruthValue	read-only	current	Y

5.11.2.9.1 adslAturIntervalNumber

Performance Data Interval number 1 is the most recent previous interval; interval 96 is 24 hours ago. Intervals 2…96 are optional.

5.11.2.9.2 adslAturIntervalLofs

Count of seconds in the interval when there was Loss of Framing.

5.11.2.9.3 adslAturIntervalLoss

Count of seconds in the interval when there was Loss of Signal.

5.11.2.9.4 adslAturIntervalLprs

Count of seconds in the interval when there was Loss of Power.

5.11.2.9.5 adslAturIntervalESs

Count of Errored Seconds in the interval. The errored second parameter is a count of one-second intervals containing one or more crc anomalies, or one or more los or sef defects.

5.11.2.9.6 adslAturIntervalValidData

This variable indicates if the data for this interval is valid.

5.11.2.10 adslAtucChanPerfDataTable

This table provides one row for each ATUC channel. ADSL channel interfaces are those ifEntries where ifType is equal to adslInterleave(124) or adslFast(125).

This table is a sequence of adslAtucChanPerfDataEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAtucChanPerfDataTable - { adslMibObjects 10 } 1.3.6.1.2.1.10.94.1.1.10
- adslAtucChanPerfDataEntry - { adslAtucChanPerfDataTable 1 } 1.3.6.1.2.1.10.94.1.1.10.1

Table 105. adslAtucChanPerfDataTable (1 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAtucChanReceivedBlks	{ adslAtucChanPerfDataEntry 1 }	Counter32	read-only	current	Y
adslAtucChanTransmittedBlks	{ adslAtucChanPerfDataEntry 2 }	Counter32	read-only	current	Y
adslAtucChanCorrectedBlks	{ adslAtucChanPerfDataEntry 3 }	Counter32	read-only	current	Y
adslAtucChanUnCorrectedBlks	{ adslAtucChanPerfDataEntry 4 }	Counter32	read-only	current	Y

Table 105. adslAtucChanPerfDataTable (2 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAtucChanPerfValidIntervals	{ adslAtucChanPerfDataEntry 5 }	INTEGER (0…96)	read-only	current	Y
adslAtucChanPerfInvalidIntervals	{ adslAtucChanPerfDataEntry 6 }	INTEGER (0…96)	read-only	current	Y
adslAtucChanPerfCurr15Min-TimeElapsed	{ adslAtucChanPerfDataEntry 7 }	AdslPerfTimeElapsed (0…899)	read-only	current	Y
adslAtucChanPerfCurr15Min-ReceivedBlks	{ adslAtucChanPerfDataEntry 8 }	PerfCurrentCount	read-only	current	Y
adslAtucChanPerfCurr15Min-Transmitted-Blks	{ adslAtucChanPerfDataEntry 9 }	PerfCurrentCount	read-only	current	Y
adslAtucChanPerfCurr15Min-CorrectedBlks	{ adslAtucChanPerfDataEntry 10 }	PerfCurrentCount	read-only	current	Y
adslAtucChanPerfCurr15Min-UncorrectBlks	{ adslAtucChanPerfDataEntry 11 }	PerfCurrentCount	read-only	current	Y
adslAtucChanPerfCurr1Day-TimeElapsed	{ adslAtucChanPerfDataEntry 12 }	AdslPerfTimeElapsed (0…86399)	read-only	current	Y
adslAtucChanPerfCurr1Day-ReceivedBlks	{ adslAtucChanPerfDataEntry 13 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucChanPerfCurr1Day-Transmitted-Blks	{ adslAtucChanPerfDataEntry 14 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucChanPerfCurr1Day-CorrectedBlks	{ adslAtucChanPerfDataEntry 15 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucChanPerfCurr1Day-UncorrectBlks	{ adslAtucChanPerfDataEntry 16 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucChanPerfPrev1Day-MoniSecs	{ adslAtucChanPerfDataEntry 17 }	INTEGER (0…86400)	read-only	current	Y
adslAtucChanPerfPrev1Day-ReceivedBlks	{ adslAtucChanPerfDataEntry 18 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucChanPerfPrev1Day-Transmitted-Blks	{ adslAtucChanPerfDataEntry 19 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucChanPerfPrev1Day-CorrectedBlks	{ adslAtucChanPerfDataEntry 20 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucChanPerfPrev1DayU-ncorrectBlks	{ adslAtucChanPerfDataEntry 21 }	AdslPerfPrevDayCount	read-only	current	Y

Note: block is intended to be the length of the channel data-block on which the CRC operates. See adslAtucChanCrcBlockLength for more information.

5.11.2.10.1 adslAtucChanReceivedBlks

Count of all encoded blocks received on this channel since agent reset.

5.11.2.10.2 adslAtucChanTransmittedBlks

Count of all encoded blocks transmitted on this channel since agent reset.

5.11.2.10.3 adslAtucChanCorrectedBlks

Count of all blocks received with errors that were corrected since agent reset. These blocks are passed on as good data.

5.11.2.10.4 adslAtucChanUncorrectBlks

Count of all blocks received with uncorrectable errors since agent reset.

5.11.2.10.5 adslAtucChanPerfValidIntervals

The number of previous 15-minute intervals in the interval table for which data was collected. Given that <n> is the maximum # of intervals supported. The value is <n> unless the measurement was (re-)started within the last (<n>*15) minutes, in which case the value is the number of complete 15 minute intervals for which the agent has at least some data. In certain cases (for example, in the case where the agent is a proxy) it is possible that some intervals are unavailable. In this case, this interval is the maximum interval number for which data is available.

5.11.2.10.6 adslAtucChanPerfInvalidIntervals

The number of intervals in the range from 0 to the value of adslAtucChanPerfValidIntervals for which no data is available. This object will typically be zero except in cases where the data for some intervals are not available (for example, in proxy situations).

5.11.2.10.7 adslAtucChanPerfCurr15MinTimeElapsed

Total elapsed seconds in this interval.

5.11.2.10.8 adslAtucChanPerfCurr15MinReceivedBlks

Count of all encoded blocks received on this channel within the current 15 minute interval.

5.11.2.10.9 adslAtucChanPerfCurr15MinTransmittedBlks

Count of all encoded blocks transmitted on this channel within the current 15 minute interval.

5.11.2.10.10 adslAtucChanPerfCurr15MinCorrectedBlks

Count of all blocks received with errors that were corrected on this channel within the current 15 minute interval.

5.11.2.10.11 adslAtucChanPerfCurr15MinUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel within the current 15 minute interval.

5.11.2.10.12 adslAtucChanPerfCurr1DayTimeElapsed

Number of seconds that have elapsed since the beginning of the current 1-day interval.

5.11.2.10.13 adslAtucChanPerfCurr1DayReceivedBlks

Count of all encoded blocks received on this channel during the current day as measured by adslAtucChanPerfCurr1DayTimeElapsed.

5.11.2.10.14 adslAtucChanPerfCurr1DayTransmittedBlks

Count of all encoded blocks transmitted on this channel during the current day as measured by adslAtucChanPerfCurr1DayTimeElapsed.

5.11.2.10.15 adslAtucChanPerfCurr1DayCorrectedBlks

Count of all blocks received with errors that were corrected on this channel during the current day as measured by adslAtucChanPerfCurr1DayTimeElapsed.

5.11.2.10.16 adslAtucChanPerfCurr1DayUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel during the current day as measured by adslAtucChanPerfCurr1DayTimeElapsed.

5.11.2.10.17 adslAtucChanPerfPrev1DayMoniSecs

The amount of time in the previous 1-day interval over which the performance monitoring information is actually counted. This value is the same as the interval duration except in a situation where performance monitoring data could not be collected for any reason.

5.11.2.10.18 adslAtucChanPerfPrev1DayReceivedBlks

Count of all encoded blocks received on this channel within the most recent previous 1-day period.

5.11.2.10.19 adslAtucChanPerfPrev1DayTransmittedBlks

Count of all encoded blocks transmitted on this channel within the most recent previous 1-day period.

5.11.2.10.20 adslAtucChanPerfPrev1DayCorrectedBlks

Count of all blocks received with errors that were corrected on this channel within the most recent previous 1-day period.

5.11.2.10.21 adslAtucChanPerfPrev1DayUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel within the most recent previous 1-day period.

5.11.2.11 adslAturChanPerfDataTable

This table provides one row for each ATUR channel. ADSL channel interfaces are those ifEntries where ifType is equal to adslInterleave(124) or adslFast(125).

This table is a sequence of adslAturChanPerfDataEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAturChanPerfDataTable - { adslMibObjects 11 } 1.3.6.1.2.1.10.94.1.1.11
- adslAturChanPerfDataEntry - { adslAturChanPerfDataTable 1 } 1.3.6.1.2.1.10.94.1.1.11.1

Table 106. adslAturChanPerfDataTable (1 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAturChanReceivedBlks	{ adslAturChanPerfDataEntry 1 }	Counter32	read-only	current	Y
adslAturChanTransmittedBlks	{ adslAturChanPerfDataEntry 2 }	Counter32	read-only	current	Y
adslAturChanCorrectedBlks	{ adslAturChanPerfDataEntry 3 }	Counter32	read-only	current	Y
adslAturChanUnCorrectedBlks	{ adslAturChanPerfDataEntry 4 }	Counter32	read-only	current	Y
adslAturChanPerfValidIntervals	{ adslAturChanPerfDataEntry 5 }	INTEGER (0…96)	read-only	current	Y
adslAturChanPerfInvalidIntervals	{ adslAturChanPerfDataEntry 6 }	INTEGER (0…96)	read-only	current	Y
adslAturChanPerfCurr15Min-TimeElapsed	{ adslAturChanPerfDataEntry 7 }	AdslPerfTimeElapsed (0…899)	read-only	current	Y
adslAturChanPerfCurr15Min-ReceivedBlks	{ adslAturChanPerfDataEntry 8 }	PerfCurrentCount	read-only	current	Y
adslAturChanPerfCurr15Min-Transmitted-Blks	{ adslAturChanPerfDataEntry 9 }	PerfCurrentCount	read-only	current	Y

Table 106. adslAturChanPerfDataTable (2 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAturChanPerfCurr15Min-CorrectedBlks	{ adslAturChanPerfDataEntry 10 }	PerfCurrentCount	read-only	current	Y
adslAturChanPerfCurr15Min-UncorrectBlks	{ adslAturChanPerfDataEntry 11 }	PerfCurrentCount	read-only	current	Y
adslAturChanPerfCurr1Day-TimeElapsed	{ adslAturChanPerfDataEntry 12 }	AdslPerfTimeElapsed (0…86399)	read-only	current	Y
adslAturChanPerfCurr1Day-ReceivedBlks	{ adslAturChanPerfDataEntry 13 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturChanPerfCurr1Day-TransmittedBlks	{ adslAturChanPerfDataEntry 14 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturChanPerfCurr1Day-CorrectedBlks	{ adslAturChanPerfDataEntry 15 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturChanPerfCurr1Day-UncorrectBlks	{ adslAturChanPerfDataEntry 16 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturChanPerfPrev1Day-MoniSecs	{ adslAturChanPerfDataEntry 17 }	INTEGER (0…86400)	read-only	current	Y
adslAturChanPerfPrev1Day-ReceivedBlks	{ adslAturChanPerfDataEntry 21 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturChanPerfPrev1Day-TransmittedBlks	{ adslAturChanPerfDataEntry 19 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturChanPerfPrev1Day-CorrectedBlks	{ adslAturChanPerfDataEntry 20 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturChanPerfPrev1Day-UncorrectBlks	{ adslAturChanPerfDataEntry 21 }	AdslPerfPrevDayCount	read-only	current	Y

Note: block is intended to be the length of the channel data-block on which the CRC operates. See adslAturChanCrcBlockLength for more information.

5.11.2.11.1 adslAturChanReceivedBlks

Count of all encoded blocks received on this channel since agent reset.

5.11.2.11.2 adslAturChanTransmittedBlks

Count of all encoded blocks transmitted on this channel since agent reset.

5.11.2.11.3 adslAturChanCorrectedBlks

Count of all blocks received with errors that were corrected since agent reset. These blocks are passed on as good data.

5.11.2.11.4 adslAturChanUncorrectBlks

Count of all blocks received with uncorrectable errors since agent reset.

5.11.2.11.5 adslAturChanPerfValidIntervals

The number of previous 15-minute intervals in the interval table for which data was collected. Given that <n> is the maximum # of intervals supported. The value is <n> unless the measurement was (re-)started within the last (<n>*15) minutes, in which case the value is the number of complete 15 minute intervals for which the agent has at least some

data. In certain cases (for example, in the case where the agent is a proxy) it is possible that some intervals are unavailable. In this case, this interval is the maximum interval number for which data is available.

5.11.2.11.6 adslAturChanPerfInvalidIntervals

The number of intervals in the range from 0 to the value of adslAturChanPerfValidIntervals for which no data is available. This object will typically be zero except in cases where the data for some intervals are not available (for example, in proxy situations).

5.11.2.11.7 adslAturChanPerfCurr15MinTimeElapsed

Total elapsed seconds in this interval. A full interval is 900 seconds.

5.11.2.11.8 adslAturChanPerfCurr15MinReceivedBlks

Count of all encoded blocks received on this channel within the current 15 minute interval.

5.11.2.11.9 adslAturChanPerfCurr15MinTransmittedBlks

Count of all encoded blocks transmitted on this channel within the current 15 minute interval.

5.11.2.11.10 adslAturChanPerfCurr15MinCorrectedBlks

Count of all blocks received with errors that were corrected on this channel within the current 15 minute interval.

5.11.2.11.11 adslAturChanPerfCurr15MinUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel within the current 15 minute interval.

5.11.2.11.12 adslAturChanPerfCurr1DayTimeElapsed

Number of seconds that have elapsed since the beginning of the current 1-day interval.

5.11.2.11.13 adslAturChanPerfCurr1DayReceivedBlks

Count of all encoded blocks received on this channel during the current day as measured by adslAturChanPerfCurr1DayTimeElapsed.

5.11.2.11.14 adslAturChanPerfCurr1DayTransmittedBlks

Count of all encoded blocks transmitted on this channel during the current day as measured by adslAturChanPerfCurr1DayTimeElapsed.

5.11.2.11.15 adslAturChanPerfCurr1DayCorrectedBlks

Count of all blocks received with errors that were corrected on this channel during the current day as measured by adslAturChanPerfCurr1DayTimeElapsed.

5.11.2.11.16 adslAturChanPerfCurr1DayUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel during the current day as measured by adslAturChanPerfCurr1DayTimeElapsed.

5.11.2.11.17 adslAturChanPerfPrev1DayMoniSecs

The amount of time in the previous 1-day interval over which the performance monitoring information is actually counted. This value is the same as the interval duration except in a situation where performance monitoring data could not be collected for any reason.

5.11.2.11.18 adslAturChanPerfPrev1DayReceivedBlks

Count of all encoded blocks received on this channel within the most recent previous 1-day period.

5.11.2.11.19 adslAturChanPerfPrev1DayTransmittedBlks

Count of all encoded blocks transmitted on this channel within the most recent previous 1-day period.

5.11.2.11.20 adslAturChanPerfPrev1DayCorrectedBlks

Count of all blocks received with errors that were corrected on this channel within the most recent previous 1-day period.

5.11.2.11.21 adslAturChanPerfPrev1DayUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel within the most recent previous 1-day period.

5.11.2.12 adslAtucChanIntervalTable

This table provides one row for each ATUC channel's performance data collection interval. ADSL channel interfaces are those ifEntries where ifType is equal to adslInterleave(124) or adslFast(125).

This table is a sequence of adslAtucChanIntervalEntry.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- adslAtucChanIntervalNumber

OIDs:

- adslAtucChanIntervalTable - { adslMibObjects 12 } 1.3.6.1.2.1.10.94.1.1.12
- adslAtucChanIntervalEntry - { adslAtucChanIntervalTable 1 } 1.3.6.1.2.1.10.94.1.1.12.1

Table 107. adslAtucChanIntervalTable

Object	OID	Syntax	Access	Status	Supported
adslAtucChanIntervalNumber	{ adslAtucChanIntervalEntry 1 }	INTEGER(1…96)	read-only	current	Y
adslAtucChanIntervalReceivedBlks	{ adslAtucChanIntervalEntry 2 }	PerfIntervalCount	read-only	current	Y
adslAtucChanIntervalTransmittedBlks	{ adslAtucChanIntervalEntry 3 }	PerfIntervalCount	read-only	current	Y
adslAtucChanIntervalCorrectedBlks	{ adslAtucChanIntervalEntry 4 }	PerfIntervalCount	read-only	current	Y
adslAtucChanIntervalUncorrectBlks	{ adslAtucChanIntervalEntry 5 }	PerfIntervalCount	read-only	current	Y
adslAtucChanIntervalValidData	{ adslAtucChanIntervalEntry 6 }	TruthValue	read-only	current	Y

5.11.2.12.1 adslAtucChanIntervalNumber

Performance Data Interval number 1 is the most recent previous interval; interval 96 is 24 hours ago. Intervals 2…96 are optional.

5.11.2.12.2 adslAtucChanIntervalReceivedBlks

Count of all encoded blocks received on this channel during this interval.

5.11.2.12.3 adslAtucChanIntervalTransmittedBlks

Count of all encoded blocks transmitted on this channel during this interval.

5.11.2.12.4 adslAtucChanIntervalCorrectedBlks

Count of all blocks received with errors that were corrected on this channel during this interval.

5.11.2.12.5 adslAtucChanIntervalUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel during this interval.

5.11.2.12.6 adslAtucChanIntervalValidData

This variable indicates if the data for this interval is valid.

5.11.2.13 adslAturChanIntervalTable

This table provides one row for each ATUR channel's performance data collection interval. ADSL channel interfaces are those ifEntries where ifType is equal to adslInterleave(124) or adslFast(125).

This table is a sequence of adslAturChanIntervalEntry.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- adslAturChanIntervalNumber

OIDs:

- adslAturChanIntervalTable - { adslMibObjects 13 } 1.3.6.1.2.1.10.94.1.1.13
- adslAturChanIntervalEntry - { adslAturChanIntervalTable 1 } 1.3.6.1.2.1.10.94.1.1.13.1

Table 108. adslAturChanIntervalTable

Object	OID	Syntax	Access	Status	Supported
adslAturChanIntervalNumber	{ adslAturChanIntervalEntry 1 }	INTEGER(1…96)	read-only	current	Y
adslAturChanIntervalReceivedBlks	{ adslAturChanIntervalEntry 2 }	PerfIntervalCount	read-only	current	Y
adslAturChanIntervalTransmittedBlks	{ adslAturChanIntervalEntry 3 }	PerfIntervalCount	read-only	current	Y
adslAturChanIntervalCorrectedBlks	{ adslAturChanIntervalEntry 4 }	PerfIntervalCount	read-only	current	Y
adslAturChanIntervalUncorrectBlks	{ adslAturChanIntervalEntry 5 }	PerfIntervalCount	read-only	current	Y
adslAturChanIntervalValidData	{ adslAturChanIntervalEntry 6 }	TruthValue	read-only	current	Y

5.11.2.13.1 adslAturChanIntervalNumber

Performance Data Interval number 1 is the most recent previous interval; interval 96 is 24 hours ago. Intervals 2…96 are optional.

5.11.2.13.2 adslAturChanIntervalReceivedBlks

Count of all encoded blocks received on this channel during this interval.

5.11.2.13.3 adslAturChanIntervalTransmittedBlks

Count of all encoded blocks transmitted on this channel during this interval.

5.11.2.13.4 adslAturChanIntervalCorrectedBlks

Count of all blocks received with errors that were corrected on this channel during this interval.

5.11.2.13.5 adslAturChanIntervalUncorrectBlks

Count of all blocks received with uncorrectable errors on this channel during this interval.

5.11.2.13.6 adslAturChanIntervalValidData

This variable indicates if the data for this interval is valid.

5.11.2.14 adslLineConfProfileTable

This table contains information on the ADSL line configuration. One entry in this table reflects a profile defined by a manager which can be used to configure the ADSL line.

This table is a sequence of adslLineConfProfileEntry. Each entry consists of a list of parameters that represents the configuration of an ADSL modem.

When “dynamic” profiles are implemented, a default profile will always exist. This profile's name is set to “DEFVAL” and its parameters are set to vendor specific values, unless otherwise specified in this document.

When “static” profiles are implemented, profiles are automatically created or destroyed as ADSL physical lines are discovered and removed by the system. The name of the profile is equivalent to the decimal value of the line's interface index.

Index:

- adslLineConfProfileName (IMPLIED)

OIDs:

- adslLineConfProfileTable - { adslMibObjects 14 } 1.3.6.1.2.1.10.94.1.1.14
- adslLineConfProfileEntry - { adslLineConfProfileTable 1 } 1.3.6.1.2.1.10.94.1.1.14.1

Table 109. adslLineConfProfileTable (1 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslLineConfProfileName	{ adslLineConfProfileEntry 1 }	SnmpAdminString (1…32)	not-accessible	current	Y
adslAtucConf-RateMode	{ adslLineConfProfileEntry 2 }	INTEGER fixed(1) adaptAtStartup(2) adaptAtRuntime(3)	read-create	current	Y
adslAtucConf-RateChanRatio	{ adslLineConfProfileEntry 3 }	INTEGER (0…100)	read-create	current	Y
adslAtucConf-TargetSnrMgn	{ adslLineConfProfileEntry 4 }	INTEGER (0…310)[a]	read-create	current	Y
adslAtucConf-MaxSnrMgn	{ adslLineConfProfileEntry 5 }	INTEGER (0…310)[a]	read-create	current	Y
adslAtucConf-MinSnrMgn	{ adslLineConfProfileEntry 6 }	INTEGER (0…310)[a]	read-create	current	Y
adslAtucConf-DownshiftSnrMgn	{ adslLineConfProfileEntry 7 }	INTEGER (0…310)[a]	read-create	current	Y
adslAtucConf-UpshiftSnrMgn	{ adslLineConfProfileEntry 8 }	INTEGER (0…310)[a]	read-create	current	Y
adslAtucConf-MinUpshiftTime	{ adslLineConfProfileEntry 9 }	INTEGER (0…16383)	read-create	current	Y
adslAtucConf-MinDownshiftTime	{ adslLineConfProfileEntry 10 }	INTEGER (0…16383)	read-create	current	Y
adslAtucChanConf-FastMinTxRate	{ adslLineConfProfileEntry 11 }	Unsigned32[b]	read-create	current	Y

Table 109. adslLineConfProfileTable (2 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAtucChanConf-InterleaveMinTxRate	{ adslLineConfProfileEntry 12 }	Unsigned32[b]	read-create	current	Y
adslAtucChanConf-FastMaxTxRate	{ adslLineConfProfileEntry 13 }	Unsigned32[b]	read-create	current	Y
adslAtucChanConf-InterleaveMaxTxRate	{ adslLineConfProfileEntry 14 }	Unsigned32[b]	read-create	current	Y
adslAtucChanConf-MaxInterleaveDelay	{ adslLineConfProfileEntry 15 }	INTEGER (0…255)	read-create	current	Y
adslAturConf-RateMode	{ adslLineConfProfileEntry 16 }	INTEGER fixed (1) adaptAtStartup (2) adaptAtRuntime (3)	read-create	current	Y
adslAturConf-RateChanRatio	{ adslLineConfProfileEntry 17 }	INTEGER (0…100)	read-create	current	Y
adslAturConf-TargetSnrMgn	{ adslLineConfProfileEntry 18 }	INTEGER (0…310)[a]	read-create	current	Y
adslAturConf-MaxSnrMgn	{ adslLineConfProfileEntry 19 }	INTEGER (0…310)[a]	read-create	current	Y
adslAturConfM-inSnrMgn	{ adslLineConfProfileEntry 20 }	INTEGER (0…310)[a]	read-create	current	Y
adslAturConf-DownshiftSnrMgn	{ adslLineConfProfileEntry 21 }	INTEGER (0…310)[a]	read-create	current	Y
adslAturConf-UpshiftSnrMgn	{ adslLineConfProfileEntry 22 }	INTEGER (0…310)[a]	read-create	current	Y
adslAturConf-MinUpshiftTime	{ adslLineConfProfileEntry 23 }	INTEGER (0…16383)	read-create	current	Y
adslAturConf-MinDownshiftTime	{ adslLineConfProfileEntry 24 }	INTEGER (0…16383)	read-create	current	Y
adslAturChanConf-FastMinTxRate	{ adslLineConfProfileEntry 25 }	Unsigned32[b]	read-create	current	Y
adslAturChanConf-InterleaveMinTxRate	{ adslLineConfProfileEntry 26 }	Unsigned32[b]	read-create	current	Y
adslAturChanConf-FastMaxTxRate	{ adslLineConfProfileEntry 27 }	Unsigned32[b]	read-create	current	Y
adslAturChanConf-InterleaveMaxTxRate	{ adslLineConfProfileEntry 28 }	Unsigned32[b]	read-create	current	Y
adslAturChanConf-MaxInterleaveDelay	{ adslLineConfProfileEntry 29 }	INTEGER (0…255)	read-create	current	Y
adslLineConf-ProfileRowStatus	{ adslLineConfProfileEntry 30 }	RowStatus	read-create	current	Y

a. Signal/Noise Margins must be entered in increments of 1dB.

b. Rates must be entered in increments of 32000 bps.

5.11.2.14.1 adslLineConfProfileName

This object is used by the line configuration table in order to identify a row of this table.

When “dynamic” profiles are implemented, the profile name is user specified. Also, the system will always provide a default profile whose name is “DEFVAL.”

When “static” profiles are implemented, there is an one-to-one relationship between each line and its profile. In which case, the profile name will need to algorithmically represent the Line's ifIndex. Therefore, the profile's name is a decimalized string of the ifIndex that is fixed-length (that is, 10) with leading zero(s). For example, the profile name for an ifIndex which equals “15” is “0000000015.”

5.11.2.14.2 adslAtucConfRateMode

Defines what form of transmit rate adaptation is configured on this modem. See ADSL Forum TR-005, “Network Management Element Management,” March 1998 for more information.

- fixed (1) - no rate adaptation
- adaptAtStartup (2) - perform rate adaptation only at initialization
- adaptAtRuntime (3) - perform rate adaptation at any time

5.11.2.14.3 adslAtucConfRateChanRatio

Configured allocation ratio of excess transmit bandwidth between fast and interleaved channels. Only applies when two channel mode and RADSL are supported. Distribute bandwidth on each channel in excess of the corresponding ChanConfMinTxRate so that:

adslAtucConfRateChanRatio =[Fast / (Fast + Interleaved)] * 100

In other words this value is the fast channel percentage.

5.11.2.14.4 adslAtucConfTargetSnrMgn

Configured Target Signal/Noise Margin. This is the Noise Margin the modem must achieve with a BER of 10-7 or better to successfully complete initialization. Expressed as in tenths of a decibel.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.5 adslAtucConfMaxSnrMgn

Configured Maximum acceptable Signal/Noise Margin. If the Noise Margin is above this the modem should attempt to reduce its power output to optimize its operation. Expressed as tenths of a decibel.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.6 adslAtucConfMinSnrMgn

Configured Minimum acceptable Signal/Noise Margin. If the noise margin falls below this level, the modem should attempt to increase its power output. If that is not possible the modem will attempt to re-initialize or shut down. Expressed as tenths of a decibel. Valid only for ReachDSL models.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.7 adslAtucConfDownshiftSnrMgn

Configured Signal/Noise Margin for rate downshift. If the noise margin falls below this level, the modem should attempt to decrease its transmit rate. In the case that RADSL mode is not present, the value is “0”. Expressed as tenths of a decibel. Valid only for ADSL models.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.8 adslAtucConfUpshiftSnrMgn

Configured Signal/Noise Margin for rate upshift. If the noise margin rises above this level, the modem should attempt to increase its transmit rate. In the case that RADSL is not present, the value is “0”. Expressed as tenths of a decibel. Valid only for ADSL models.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.9 adslAtucConfMinUpshiftTime

Minimum time, in seconds, that the current margin is above UpshiftSnrMgn before an upshift occurs. In the case that RADSL is not present, the value is “0.”

5.11.2.14.10 adslAtucConfMinDownshiftTime

Minimum time, in seconds, that the current margin is below DownshiftSnrMgn before a downshift occurs. In the case that RADSL mode is not present, the value is “0.” Valid only for ADSL models.

5.11.2.14.11 adslAtucChanConfFastMinTxRate

Configured Minimum Transmit rate for “Fast” channels, in bps. See adslAtucConfRateChanRatio for information regarding RADSL mode and ATUR transmit rate for ATUC receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- All values must be in increments of 32 000bps
- G.dmt, ANSI, and multimode
  - — 64 000 to 12 000 000
- G.lite
  - — 64 000 to 4 000 000
- ReachDSL
  - — 32 000 to 2 176 000

5.11.2.14.12 adslAtucChanConfInterleaveMinTxRate

Configured Minimum Transmit rate for “Interleave” channels, in bps. See adslAtucConfRateChanRatio for information regarding RADSL mode and see ATUR transmit rate for receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, and multimode
  - — 64 000 to 12 000 000
- G.lite
  - — 64 000 to 4 000 000

5.11.2.14.13 adslAtucChanConfFastMaxTxRate

Configured Maximum Transmit rate for “Fast” channels, in bps. See adslAtucConfRateChanRatio for information regarding RADSL mode and see ATUR transmit rate for ATUC receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, and multimode
  - — 64 000 to 12 000 000
- G.lite
  - — 64 000 to 4 000 000
- ReachDSL

— 32 000 to 2 176 000

5.11.2.14.14 adslAtucChanConfInterleaveMaxTxRate

Configured Maximum Transmit rate for "Interleave" channels, in bps. See adslAtucConfRateChanRatio for information regarding RADSL mode and ATUR transmit rate for ATUC receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, and multimode
  — 64 000 to 12 000 000
- G.lite
  — 64 000 to 4 000 000

5.11.2.14.15 adslAtucChanConfMaxInterleaveDelay

Configured maximum Interleave Delay for this channel.

Interleave delay, in milli-seconds, applies only to the interleave channel and defines the mapping (relative spacing) between subsequent input bytes at the interleaver input and their placement in the bit stream at the interleaver output. Larger numbers provide greater separation between consecutive input bytes in the output bit stream allowing for improved impulse noise immunity at the expense of payload latency.

5.11.2.14.16 adslAturConfRateMode

Defines what form of transmit rate adaptation is configured on this modem. See ADSL Forum TR-005, "Network Management Element Management," March 1998 for more information.

- fixed (1) - no rate adaptation
- adaptAtStartup (2) - perform rate adaptation only at initialization
- adaptAtRuntime (3) - perform rate adaptation at any time

5.11.2.14.17 adslAturConfRateChanRatio

Configured allocation ratio of excess transmit bandwidth between fast and interleaved channels. Only applies when two channel mode and RADSL are supported. Distribute bandwidth on each channel in excess of the corresponding ChanConfMinTxRate so that:

adslAturConfRateChanRatio = [Fast / (Fast + Interleaved)] * 100

In other words this value is the fast channel percentage.

5.11.2.14.18 adslAturConfTargetSnrMgn

Configured Target Signal/Noise Margin. This is the Noise Margin the modem must achieve with a BER of 10-7 or better to successfully complete initialization.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.19 adslAturConfMaxSnrMgn

Configured Maximum acceptable Signal/Noise Margin. If the Noise Margin is above this the modem should attempt to reduce its power output to optimize its operation. Expressed as tenths of a decibel.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.20 adslAturConfMinSnrMgn

Configured Minimum acceptable Signal/Noise Margin. If the noise margin falls below this level, the modem should attempt to increase its power output. If that is not possible the modem will attempt to re-initialize or shut down. Expressed as tenths of a decibel.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.21 adslAturConfDownshiftSnrMgn

Configured Signal/Noise Margin for rate downshift. If the noise margin falls below this level, the modem should attempt to decrease its transmit rate. In the case that RADSL mode is not present, the value is “0”. Expressed as tenths of a decibel.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.22 adslAturConfUpshiftSnrMgn

Configured Signal/Noise Margin for rate upshift. If the noise margin rises above this level, the modem should attempt to increase its transmit rate. In the case that RADSL is not present, the value is “0”. Expressed as tenths of a decibel.

Valid Object Value(s)

Values must be in 1dB increments.

5.11.2.14.23 adslAturConfMinUpshiftTime

Minimum time, in seconds, that the current margin is above UpshiftSnrMgn before an upshift occurs. In the case that RADSL is not present, the value is “0.”

5.11.2.14.24 adslAturConfMinDownshiftTime

Minimum time, in seconds, that the current margin is below DownshiftSnrMgn before a downshift occurs. In the case that RADSL mode is not present, the value is “0.”

5.11.2.14.25 adsladslAturChanConfFastMinTxRate

Configured Minimum Transmit rate for “Fast” channels, in bps. See adslAturConfRateChanRatio for information regarding RADSL mode and ATUC transmit rate for ATUR receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, G.lite, and multimode
  — 64 000 to 1 024 000
- ReachDSL
  — 32 000 to 2 176 000

5.11.2.14.26 adslAturChanConfInterleaveMinTxRate

Configured Minimum Transmit rate for “Interleave” channels, in bps. See adslAturConfRateChanRatio for information regarding RADSL mode and ATUC transmit rate for ATUR receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, G.lite, and multimode
  — 64 000 to 1 024 000

5.11.2.14.27 adslAturChanConfFastMaxTxRate

Configured Maximum Transmit rate for “Fast” channels, in bps. See adslAturConfRateChanRatio for information regarding RADSL mode and ATUC transmit rate for ATUR receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, G.lite, and multimode

— 64 000 to 1 024 000

- ReachDSL
  — 32 000 to 2 176 000

5.11.2.14.28 adslAturChanConfInterleaveMaxTxRate

Configured Maximum Transmit rate for “Interleave” channels, in bps. See adslAturConfRateChanRatio for information regarding RADSL mode and see ATUC transmit rate for ATUR receive rates.

Valid Object Value(s)

Note: Spaces added in values to improve readability. They are not actually part of the entry.

- G.dmt, ANSI, G.lite, and multimode
  — 64 000 to 1 024 000

5.11.2.14.29 adslAturChanConfMaxInterleaveDelay

Configured maximum Interleave Delay, in milli-seconds, for this channel.

Interleave delay applies only to the interleave channel and defines the mapping (relative spacing) between subsequent input bytes at the interleaver input and their placement in the bit stream at the interleaver output. Larger numbers provide greater separation between consecutive input bytes in the output bit stream allowing for improved impulse noise immunity at the expense of payload latency.

5.11.2.14.30 adslLineConfProfileRowStatus

This object is used to create a new row or modify or delete an existing row in this table.

A profile activated by setting this object to “active.” When “active” is set, the system will validate the profile.

Before a profile can be deleted or taken out of service, (by setting this object to “destroy” or “outOfService”) it must be first unreferenced from all associated lines.

If the implementor of this MIB has chosen not to implement “dynamic assignment” of profiles, this object's MIN-ACCESS is read-only and its value is always to be “active.”

Valid Object Value(s)

The createAndWait RowStatus (Section 4.6, Page 13) is not supported.

5.11.2.15 adslLineAlarmConfProfileTable

This table contains information on the ADSL line configuration. One entry in this table reflects a profile defined by a manager which can be used to configure the modem for a physical line

This table is a sequence of adslLineAlarmConfProfileEntry.

Each entry consists of a list of parameters that represents the configuration of an ADSL modem.

When “dynamic” profiles are implemented, a default profile will always exist. This profile's name is set to “DEFVAL” and its parameters are set to vendor specific values, unless otherwise specified in this document.

When “static” profiles are implemented, profiles are automatically created or destroyed as ADSL physical lines are discovered and removed by the system. The name of the profile is equivalent to the decimal value of the line's interface index.

Index:

- adslLineAlarmConfProfileName (IMPLIED)

OIDs:

- adslLineAlarmConfProfileTable - { adslMibObjects 15 } 1.3.6.1.2.1.10.94.1.1.15
- adslLineAlarmConfProfileEntry - { adslLineAlarmConfProfileTable 1 } 1.3.6.1.2.1.10.94.1.1.15.1

Table 110. adslLineAlarmConfProfileTable

Object	OID	Syntax	Access	Status	Sup-ported
adslLineAlarmConf-ProfileName	{ adslLineAlarmConfProfileEntry 1 }	SnmpAdminString (SIZE (1…32))	not-accessible	current	Y
adslAtucThresh15Min-Lofs	{ adslLineAlarmConfProfileEntry 2 }	INTEGER (0…900)	read-create	current	Y
adslAtucThresh15Min-Loss	{ adslLineAlarmConfProfileEntry 3 }	INTEGER (0…900)	read-create	current	Y
adslAtucThresh15Min-Lols	{ adslLineAlarmConfProfileEntry 4 }	INTEGER (0…900)	read-create	current	Y
adslAtucThresh15Min-Lprs	{ adslLineAlarmConfProfileEntry 5 }	INTEGER (0…900)	read-create	current	Y
adslAtucThresh15Min-ESs	{ adslLineAlarmConfProfileEntry 6 }	INTEGER (0…900)	read-create	current	Y
adslAtucThreshFast-RateUp	{ adslLineAlarmConfProfileEntry 7 }	Unsigned32	read-create	current	Y
adslAtucThreshInterleave-RateUp	{ adslLineAlarmConfProfileEntry 8 }	Unsigned32	read-create	current	Y
adslAtucThreshFast-RateDown	{ adslLineAlarmConfProfileEntry 9 }	Unsigned32	read-create	current	Y
adslAtucThreshInterleave-RateDown	{ adslLineAlarmConfProfileEntry 10 }	Unsigned32	read-create	current	Y
adslAtucInitFailure-TrapEnable	{ adslLineAlarmConfProfileEntry 11 }	INTEGER enable (1) disable (2)	read-create	current	Y
adslAturThresh15Min-Lofs	{ adslLineAlarmConfProfileEntry 12 }	INTEGER (0…900)	read-create	current	Y
adslAturThresh15Min-Loss	{ adslLineAlarmConfProfileEntry 13 }	INTEGER (0…900)	read-create	current	Y
adslAturThresh15Min-Lprs	{ adslLineAlarmConfProfileEntry 14 }	INTEGER (0…900)	read-create	current	Y
adslAturThresh15Min-ESs	{ adslLineAlarmConfProfileEntry 15 }	INTEGER (0…900)	read-create	current	Y
adslAturThreshFastRate-Up	{ adslLineAlarmConfProfileEntry 16 }	Unsigned32	read-create	current	Y
adslAturThreshInterleave-RateUp	{ adslLineAlarmConfProfileEntry 17 }	Unsigned32	read-create	current	Y
adslAturThreshFast-RateDown	{ adslLineAlarmConfProfileEntry 18 }	Unsigned32	read-create	current	Y
adslAturThreshInterleave-RateDown	{ adslLineAlarmConfProfileEntry 19 }	Unsigned32	read-create	current	Y
adslAturAlarmConfProfile-RowStatus	{ adslLineAlarmConfProfileEntry 20 }	RowStatus	read-create	current	Y

5.11.2.15.1 adslLineAlarmConfProfileName

This object is used by the line alarm configuration table in order to identify a row of this table.

When “dynamic” profiles are implemented, the profile name is user specified. Also, the system will always provide a default profile whose name is “DEFVAL.”

When “static” profiles are implemented, there is an one-to-one relationship between each line and its profile. In which case, the profile name will need to algorithmically represent the Line's ifIndex. Therefore, the profile's name is a decimalized string of the ifIndex that is fixed-length (that is, 10) with leading zero(s). For example, the profile name for an ifIndex which equals '15' is '0000000015'.

5.11.2.15.2 adslAtucThresh15MinLofs

The number of Loss of Frame Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAtucPerfLofsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap. This value is expressed in units of seconds.

5.11.2.15.3 adslAtucThresh15MinLoss

The number of Loss of Signal Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAtucPerfLossThreshTrap. One trap is sent per interval per interface. A value of “0’ will disable the trap.

5.11.2.15.4 adslAtucThresh15MinLols

The number of Loss of Link Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAtucPerfLolsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.5 adslAtucThresh15MinLprs

The number of Loss of Power Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAtucPerfLprsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.6 adslAtucThresh15MinESs

The number of Errored Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAtucPerfESsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.7 adslAtucThreshFastRateUp

Applies to “Fast” channels only. Configured change in rate, in bps, causing an adslAtucRateChangeTrap. A trap is produced when: ChanCurrTxRate >= ChanPrevTxRate plus the value of this object. A value of “0” will disable the trap.

5.11.2.15.8 adslAtucThreshInterleaveRateUp

Applies to “Interleave” channels only. Configured change in rate, in bps, causing an adslAtucRateChangeTrap. A trap is produced when: ChanCurrTxRate >= ChanPrevTxRate plus the value of this object. A value of “0” will disable the trap.

5.11.2.15.9 adslAtucThreshFastRateDown

Applies to “Fast” channels only. Configured change in rate, in bps, causing an adslAtucRateChangeTrap. A trap is produced when: ChanCurrTxRate <= ChanPrevTxRate minus the value of this object. A value of “0” will disable the trap.

5.11.2.15.10 adslAtucThreshInterleaveRateDown

Applies to “Interleave” channels only. Configured change in rate, in bps, causing an adslAtucRateChangeTrap. A trap is produced when: ChanCurrTxRate <= ChanPrevTxRate minus the value of this object. A value of “0” will disable the trap.

5.11.2.15.11 adslAtucInitFailureTrapEnable

Enables and disables the InitFailureTrap. This object is defaulted disable(2)

- enable (1)
- disable (2)

5.11.2.15.12 adslAturThresh15MinLofs

The number of Loss of Frame Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAturPerfLofsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.13 adslAturThresh15MinLoss

The number of Loss of Signal Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAturPerfLossThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.14 adslAturThresh15MinLprs

The number of Loss of Power Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAturPerfLprsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.15 adslAturThresh15MinESs

The number of Errored Seconds encountered by an ADSL interface within any given 15 minutes performance data collection period, which causes the SNMP agent to send an adslAturPerfESsThreshTrap. One trap is sent per interval per interface. A value of “0” will disable the trap.

5.11.2.15.16 adslAturThreshFastRateUp

Applies to “Fast” channels only. Configured change in rate, in bps, causing an adslAturRateChangeTrap. A trap is produced when: ChanCurrTxRate >= ChanPrevTxRate plus the value of this object. A value of “0” will disable the trap.

5.11.2.15.17 adslAturThreshInterleaveRateUp

Applies to “Interleave” channels only. configured change in rate, in bps, causing an adslAturRateChangeTrap. A trap is produced when: ChanCurrTxRate >= ChanPrevTxRate plus the value of this object. A value of “0” will disable the trap.

5.11.2.15.18 adslAturThreshFastRateDown

Applies to “Fast” channels only. Configured change in rate, in bps, causing an adslAturRateChangeTrap. A trap is produced when: ChanCurrTxRate <= ChanPrevTxRate minus the value of this object. A value of “0” will disable the trap.

5.11.2.15.19 adslAturThreshInterleaveRateDown

Applies to “Interleave” channels only. Configured change in rate, in bps, causing an adslAturRateChangeTrap. A trap is produced when: ChanCurrTxRate <= ChanPrevTxRate minus the value of this object. A value of “0” will disable the trap.

5.11.2.15.20 adslLineAlarmConfProfileRowStatus

This object is used to create a new row or modify or delete an existing row in this table.

A profile activated by setting this object to “active.” When “active” is set, the system will validate the profile.

Before a profile can be deleted or taken out of service, (by setting this object to “destroy” or “outOfService”) it must be first unreferenced from all associated lines.

If the implementor of this MIB has chosen not to implement “dynamic assignment” of profiles, this object's MIN-ACCESS is read-only and its value is always to be “active”.

5.11.2.16 adslLCSMib

This section is a place holder for the Line Code Specific MIB objects once they are defined. Line Code Specific MIB objects may be defined in other RFCs or TR type of documents.

OIDs:

- adslLCSMib - { adslMibObjects 16 } 1.3.6.1.2.1.10.94.1.1.16

5.11.3 Traps

This section defines the traps generated by an ADSL line interface.

OIDs:

- adslTraps - { adslLineMib 2 } 1.3.6.1.2.1.10.94.2
- adslAtucTraps - { adslTraps 1 } 1.3.6.1.2.1.10.94.2.1
- adslAturTraps - { adslTraps 2 } 1.3.6.1.2.1.10.94.2.2

5.11.3.1 adslAtucTraps

Table 111. adslAtucTraps

Notification	OID	Variable Bindings	Status	Supported
adslAtucPerfLofsThreshTrap	{ adslAtucTraps 0 1 }	adslAtucPerfCurr15MinLofs adslAtucThresh15MinLofs	current	Y
adslAtucPerfLossThreshTrap	{ adslAtucTraps 0 2 }	adslAtucPerfCurr15MinLoss adslAtucThresh15MinLoss	current	Y
adslAtucPerfLprsThreshTrap	{ adslAtucTraps 0 3 }	adslAtucPerfCurr15MinLprs adslAtucThresh15MinLprs	current	N
adslAtucPerfESsTreshTrap	{ adslAtucTraps 0 4 }	adslAtucPerfCurr15MinESs adslAtucThresh15MinESs	current	Y
adslAtucRateChangeTrap	{ adslAtucTraps 0 5 }	adslAtucChanCurrTxRate adslAtucChanPrevTxRate	current	Y
adslAtucPerfLolsThreshTrap	{ adslAtucTraps 0 6 }	adslAtucPerfCurr15MinLols adslAtucThresh15MinLols	current	Y
adslAtucInitFailureTrap	{ adslAtucTraps 0 7 }	adslAtucCurrStatus	current	Y

5.11.3.1.1 adslAtucPerfLofsThreshTrap

Loss of Framing 15-minute interval threshold reached.

5.11.3.1.2 adslAtucPerfLossThreshTrap

Loss of Signal 15-minute interval threshold reached.

5.11.3.1.3 adslAtucPerfLprsThreshTrap

Loss of Power 15-minute interval threshold reached.

5.11.3.1.4 adslAtucPerfESsThreshTrap

Errored Second 15-minute interval threshold reached.

5.11.3.1.5 adslAtucRateChangeTrap

The ATUCs transmit rate has changed (RADSL mode only)

5.11.3.1.6 adslAtucPerfLolsThreshTrap

Loss of Link 15-minute interval threshold reached.

5.11.3.1.7 adslAtucInitFailureTrap

ATUC initialization failed. See adslAtucCurrStatus for potential reasons.

5.11.3.2 adslAturTraps

Table 112. adslAturTraps

Notification	OID	Variable Bindings	Status	Supported
adslAturPerfLofsThreshTrap	{ adslAturTraps 0 1 }	adslAturPerfCurr15MinLofs adslAturThresh15MinLofs	current	Y
adslAturPerfLossThreshTrap	{ adslAturTraps 0 2 }	adslAturPerfCurr15MinLoss adslAturThresh15MinLoss	current	Y
adslAturPerfLprsThreshTrap	{ adslAturTraps 0 3 }	adslAtucPerfCurr15MinLprs adslAturThresh15MinLprs	current	Y
adslAturPerfESsTreshTrap	{ adslAturTraps 0 4 }	adslAturPerfCurr15MinESs adslAturThresh15MinESs	current	Y
adslAturRateChangeTrap	{ adslAturTraps 0 5 }	adslAturChanCurrTxRate adslAturChanPrevTxRate	current	Y

5.11.3.2.1 adslAturPerfLofsThreshTrap

Loss of Framing 15-minute interval threshold reached.

5.11.3.2.2 adslAturPerfLossThreshTrap

Loss of Signal 15-minute interval threshold reached.

5.11.3.2.3 adslAturPerfLprsThreshTrap

Loss of Power 15-minute interval threshold reached.

5.11.3.2.4 adslAturPerfESsThreshTrap

Errored Second 15-minute interval threshold reached.

5.11.3.2.5 adslAturRateChangeTrap

The ATURs transmit rate has changed (RADSL mode only)

5.12 ADSL-LINE-EXT-MIB — RFC 3440

5.12.1 Introduction

This MIB serves as an extension to the ADSL Line MIB (Section 5.11, Page 173) and describes additional objects for managing ADSL interfaces not covered by the ADSL Line MIB. The document used for the ADSL-LINE-EXT-MIB was dated December 10, 2002 and has a LAST-UPDATED of "200212100000Z."

5.12.2 Textual Conventions

The following textual conventions are made in the reference document:

- AdslTransmissionModeType - A set of ADSL line transmission modes, with one bit per mode. The notes (F) and (L) denote Full-Rate and G.Lite respectively.
  - SYNTAX BITS
    - ansit1413(0) : Bit 00 : Regional Std. (ANSI T1.413) (F)
    - etsi(1) : Bit 01 : Regional Std. (ETSI DTS/TM06006) (F)
    - q9921PotsNonOverlapped(2) : Bit 02 : G.992.1 POTS non-overlapped (F)
    - q9921PotsOverlapped(3) : Bit 03 : G.992.1 POTS overlapped (F)
    - q9921IsdnNonOverlapped(4) : Bit 04 : G.992.1 ISDN non-overlapped (F)
    - q9921isdnOverlapped(5) : Bit 05 : G.992.1 ISDN overlapped (F)
    - q9921tcmIsdnNonOverlapped(6) : Bit 06 : G.992.1 TCM-ISDN non-overlapped (F)
    - q9921tcmIsdnOverlapped(7) : Bit 07 : G.992.1 TCM-ISDN overlapped (F)
    - q9922potsNonOverlapped(8) : Bit 08 : G.992.2 POTS non-overlapped (L)
    - q9922potsOverlapped(9) : Bit 09 : G.992.2 POTS overlapped (L)
    - q9922tcmIsdnNonOverlapped(10) : Bit 10 : G.992.2 with TCM-ISDN non-overlapped (L)
    - q9922tcmIsdnOverlapped(11) : Bit 11 : G.992.2 with TCM-ISDN overlapped (L)
    - q9921tcmIsdnSymmetric(12) : Bit 12 : G.992.1 TCM-ISDN symmetric (F)

5.12.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- transmission - { mib-2 10 } 1.3.6.1.2.1.10
- adslMIB - { transmission 94 } 1.3.6.1.2.1.10.94
- adslExtMIB - { adslMIB 3 } 1.3.6.1.2.1.10.94.3
- adslExtMibObjects - { adslExtMIB 1 } 1.3.6.1.2.1.10.94.3.1

5.12.4 MIB Objects

The table below identifies the objects called out in this MIB. Any table object that has elements that are supported are indicated by a "Y" in the Supported column.

Table 113. ADSL-LINE-EXT-MIB Objects (1 of 2)

Object	OID	Syntax SEQUENCE of	Access	Status	Sup-ported
adslLineExtTable	{ adslMibObjects 17 }	adslLineExtEntry	not-accessible	current	Y
adslAtucPerfDataExtTable	{ adslMibObjects 18 }	adslAtucPerfDataExtEntry	not-accessible	current	Y

Table 113. ADSL-LINE-EXT-MIB Objects (2 of 2)

Object	OID	Syntax SEQUENCE of	Access	Status	Sup-ported
adslAtucIntervalExtTable	{ adslMibObjects 19 }	adslAtucIntervalExtEntry	not-accessible	current	Y
adslAturPerfDataExtTable	{ adslMibObjects 20 }	adslAturPerfDataExtEntry	not-accessible	current	Y
adslAturIntervalExtTable	{ adslMibObjects 21 }	adslAturIntervalExtEntry	not-accessible	current	Y
adslConfProfileExtTable	{ adslMibObjects 22 }	adslConfProfileExtEntry	not-accessible	current	Y
adslAlarmConfProfileExtTable	{ adslMibObjects 23 }	adslAlarmConfProfileExtEntry	not-accessible	current	Y

5.12.4.1 adslLineExtTable

This table is an extension of the adslLineTable RFC 2662. It contains ADSL line configuration and monitoring information. This includes the ADSL line's capabilities and actual ADSL transmission system. Each entry corresponds to an ADSL line.

Augments:

- adslLineTable (Section 5.11.2.1, Page 175)

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslLineExtTable - { adslExtMibObjects 17 } 1.3.6.1.2.1.10.94.3.1.17
- adslLineExtEntry - { adslLineExtTable 1 } 1.3.6.1.2.1.10.94.3.1.17.1

Table 114. adslLineExtTable

Object	OID	Syntax	Access	Status	Supported
adslLineTransAtucCap	{ adslLineExtEntry 1 }	AdslTransmissionModeType	read-only	current	Y
adslLineTransAtucConfig	{ adslLineExtEntry 2 }	AdslTransmissionModeType	read-write	current	Y
adslLineTransAtucActual	{ adslLineExtEntry 3 }	AdslTransmissionModeType	read-only	current	Y
adslLineGlitePowerState	{ adslLineExtEntry 4 }	INTEGER none(1) l0(2) l1(3) l3(4)	read-only	current	Y
adslLineConfProfileDualLite	{ adslLineExtEntry 5 }	SnmpAdminString[a]	read-write	current	N

a. The SnmpAdminString syntax for this object has not been restricted in size as the adslLineConfProfile (Section 5.11.2.1.4, Page 177) object in the adslLineTable (Section 5.11.2.1, Page 175). Since the adslLineConfigProfile object is restricted to maximum of 32 bytes, this object will assume the restriction as well.

5.12.4.1.1 adslLineTransAtucCap

The transmission modes, represented by a bitmask that the ATU-C is capable of supporting. The modes available are limited by the design of the equipment.

Valid Object Value(s)

This object is set to a bitwise OR indicating the following capabilities:

Table 115. adslLineTransAtucCap for Annex A Model

AdslTransmissionModeType	Description	Value (hex)
ansit1413(0)	ANSI T1.413	80 00
q9921PotsNonOverlapped(2)	G.992.1 POTS non-overlapped Annex A	20 00
q9922PostNonOverlapped(8)	G.992.2 POTS non-overlapped Annex A	00 80

Table 116. adslLineTransAtucCap for Annex B

AdslTransmissionModeType	Description	Value (hex)
q9921IsdnNonOverlapped(4))	G.992.1 ISDN non-overlapped Annex B	08 00

5.12.4.1.2 adslLineTransAtucConfig

The transmission modes, represented by a bitmask, currently enabled by the ATU-C. The manager can only set those modes that are supported by the ATU-C. An ATU-C's supported modes are provided by adslLineTransAtucCap (Section 5.12.4.1.1, Page 212).

5.12.4.1.3 adslLineTransAtucActual

The actual transmission mode of the ATU-C. During ADSL line initialization, the ADSL Transceiver Unit - Remote terminal end (ATU-R) will determine the mode used for the link. This value is limited to a single transmission mode that is a subset of those modes enabled by the ATU-C and denoted by adslLineTransAtucConfig (Section 5.12.4.1.2, Page 213). After an initialization has occurred, its mode is saved as the "Current" mode and is persistence should the link go down. This object returns 0 (that is, BITS [Section 4.1, Page 11] with no mode bit set) if the mode is not known.

5.12.4.1.4 adslLineGlitePowerState

The value of this object specifies the power state of this interface. The following are valid values for this object:

- none(1)
- l0(2) - L0 Power on
- l1(3) - L1 Power on but reduced
- l3(4) - L3 Power off

The value of this object specifies the power state of this interface. L0 is power on, L1 is power on but reduced and L3 is power off. Power state cannot be configured by an operator but it can be viewed via the ifOperStatus (Section 5.3.4.3.8, Page 46) object for the managed ADSL interface. The value of the object ifOperStatus is set to down(2) if the ADSL interface is in power state L3 and is set to up(1) if the ADSL line interface is in power state L0 or L1. If the object adslLineTransAtucActual (Section 5.12.4.1.3, Page 213) is set to a G.992.2 (G.Lite)-type transmission mode, the value of this object is one of the valid power states: L0(2), L1(3), or L3(4). Otherwise, its value is none(1).

5.12.4.1.5 adslLineConfProfileDualLite

This object extends the definition an ADSL line and associated channels (when applicable) for cases when it is configured in dual mode, and operating in a G.Lite-type mode as denoted by adslLineTransAtucActual (Section 5.12.4.1.3, Page 213). Dual mode exists when the object, adslLineTransAtucConfig (Section 5.12.4.1.3, Page 213) is configured with one or more full-rate modes and one or more G.Lite modes simultaneously.

When “dynamic” profiles are implemented, the value of object is equal to the index of the applicable row in the ADSL Line Configuration Profile Table, adslLineConfProfileTable (Section 5.11.2.14, Page 198) defined in ADSL-MIB [RFC 2662].

In the case when dual-mode has not been enabled, the value of the object is equal to the value of the object adslLineConfProfile (Section 5.11.2.1.4, Page 177) [RFC 2662].

When “static” profiles are implemented, in much like the case of the object, adslLineConfProfileName (Section 5.11.2.14.1, Page 200) [RFC 2662], this object’s value will need to algorithmically represent the characteristics of the line. In this case, the value of the line’s ifIndex plus a value indicating the line mode type (for example, G.Lite, Full-rate) is used. Therefore, the profile’s name is a string concatenating the ifIndex (Section 5.3.4.3.1, Page 37) and one of the follow values: Full or Lite. This string is fixed-length (that is, 14) with leading zero(s). For example, the profile name for an ifIndex that equals “15” and is a full rate line is 0000000015Full.

5.12.4.2 adslAtucPerfDataExtTable

This table extends adslAtucPerfDataTable (Section 5.11.2.6, Page 182) [RFC 2662] with additional ADSL physical line counter information such as unavailable seconds-line and severely errored seconds-line. Each entry corresponds to an ADSL line.

Augments:

- adslAtucPerfDataTable (Section 5.11.2.6, Page 182)

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAtucPerfDataExtTable - { adslExtMibObjects 18 } 1.3.6.1.2.1.10.94.3.1.18
- adslAtucPerfDataExtEntry - { adslAtucPerfDataExtTable 1 } 1.3.6.1.2.1.10.94.3.1.18.1

Table 117. adslAtucPerfDataExtTable (1 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAtucPerfStatFastR	{ adslAtucPerfDataExtEntry 1 }	Counter32	read-only	current	Y
adslAtucPerfStatFailedFastR	{ adslAtucPerfDataExtEntry 2 }	Counter32	read-only	current	Y
adslAtucPerfStatSesL	{ adslAtucPerfDataExtEntry 3 }	Counter32	read-only	current	Y
adslAtucPerfStatUasL	{ adslAtucPerfDataExtEntry 4 }	Counter32	read-only	current	Y
adslAtucPerfCurr15MinFastR	{ adslAtucPerfDataExtEntry 5 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinFailedFastR	{ adslAtucPerfDataExtEntry 6 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinSesL	{ adslAtucPerfDataExtEntry 7 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr15MinUasL	{ adslAtucPerfDataExtEntry 8 }	PerfCurrentCount	read-only	current	Y
adslAtucPerfCurr1DayFastR	{ adslAtucPerfDataExtEntry 9 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayFailedFastR	{ adslAtucPerfDataExtEntry 10 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DaySesL	{ adslAtucPerfDataExtEntry 11 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfCurr1DayUasL	{ adslAtucPerfDataExtEntry 12 }	AdslPerfCurrDayCount	read-only	current	Y
adslAtucPerfPrev1DayFastR	{ adslAtucPerfDataExtEntry 13 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DayFailedFastR	{ adslAtucPerfDataExtEntry 14 }	AdslPerfPrevDayCount	read-only	current	Y
adslAtucPerfPrev1DaySesL	{ adslAtucPerfDataExtEntry 15 }	AdslPerfPrevDayCount	read-only	current	Y

Table 117. adslAtucPerfDataExtTable (2 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
adslAtucPerfPrev1DayUasL	{ adslAtucPerfDataExtEntry 16 }	AdslPerfPrevDayCount	read-only	current	Y

5.12.4.2.1 adslAtucPerfStatFastR

The value of this object reports the count of the number of fast line retrains since last agent reset.

5.12.4.2.2 adslAtucPerfStatFailedFastR

The value of this object reports the count of the number of failed fast line retrains since last agent reset.

The value of this object indicates the count of failed fast retrains.

5.12.4.2.3 adslAtucPerfStatSesL

The value of this object reports the count of the number of severely errored seconds-line since last agent reset.

The value of this object indicates the count of severely errored second-line.

5.12.4.2.4 adslAtucPerfStatUasL

The value of this object reports the count of the number of unavailable seconds-line since last agent reset.

The value of this object indicates the count of unavailable second.

5.12.4.2.5 adslAtucPerfCurr15MinFastR

For the current 15-minute interval, this object reports the current number of seconds during which there have been fast retrains.

5.12.4.2.6 adslAtucPerfCurr15MinFailedFastR

For the current 15-minute interval, this object reports the current number of seconds during which there have been failed fast retrains.

5.12.4.2.7 adslAtucPerfCurr15MinSesL

For the current 15-minute interval, this object reports the current number of seconds during which there have been severely errored seconds-line.

5.12.4.2.8 adslAtucPerfCurr15MinUasL

For the current 15-minute interval, this object reports the current number of seconds during which there have been unavailable seconds-line.

5.12.4.2.9 adslAtucPerfCurr1DayFastR

For the current day as measured by adslAtucPerfCurr1DayTimeElapsed (Section 5.11.2.6.16, Page 184) [RFC 2662], this object reports the number of seconds during which there have been fast retrains.

5.12.4.2.10 adslAtucPerfCurr1DayFailedFastR

For the current day as measured by adslAtucPerfCurr1DayTimeElapsed (Section 5.11.2.6.16, Page 184) [RFC 2662], this object reports the number of seconds during which there have been failed fast retrains.

5.12.4.2.11 adslAtucPerfCurr1DaySesL

For the current day as measured by adslAtucPerfCurr1DayTimeElapsed (Section 5.11.2.6.16, Page 184) [RFC 2662], this object reports the number of seconds during which there have been severely errored seconds-line.

5.12.4.2.12 adslAtucPerfCurr1DayUasL

For the current day as measured by adslAtucPerfCurr1DayTimeElapsed (Section 5.11.2.6.16, Page 184) [RFC 2662], this object reports the number of seconds during which there have been unavailable seconds-line.

5.12.4.2.13 adslAtucPerfPrev1DayFastR

For the previous day, this object reports the number of seconds during which there were fast retrains.

5.12.4.2.14 adslAtucPerfPrev1DayFailedFastR

For the previous day, this object reports the number of seconds during which there were failed fast retrains.

5.12.4.2.15 adslAtucPerfPrev1DaySesL

For the previous day, this object reports the number of seconds during which there were severely errored seconds-line.

5.12.4.2.16 adslAtucPerfPrev1DayUasL

For the previous day, this object reports the number of seconds during which there were unavailable seconds-line.

5.12.4.3 adslAtucIntervalExtTable

This table provides one row for each ATU-C performance data collection interval for ADSL physical interfaces whose IfEntries' ifType (Section 5.3.4.3.3, Page 41) is equal to adsl(94).

Augments:

- adslAtucIntervalTable (Section 5.11.2.8, Page 188)

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- adslAtucIntervalNumber (Section 5.11.2.8.1, Page 189)

OIDs:

- adslAtucIntervalExtTable - { adslExtMibObjects 19 }
- adslAtucIntervalExtEntry - { adslAtucIntervalExtTable 1 }

Table 118. adslAtucIntervalExtTable

Object	OID	Syntax	Access	Status	Supported
adslAtucIntervalFastR	{ adslAtucIntervalExtEntry 1 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalFailedFastR	{ adslAtucIntervalExtEntry 2 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalSesL	{ adslAtucIntervalExtEntry 3 }	PerfIntervalCount	read-only	current	Y
adslAtucIntervalUasL	{ adslAtucIntervalExtEntry 4 }	PerfIntervalCount	read-only	current	Y

5.12.4.3.1 adslAtucIntervalFastR

For the current interval, this object reports the current number of seconds during which there have been fast retrains.

5.12.4.3.2 adslAtucIntervalFailedFastR

For the each interval, this object reports the number of seconds during which there have been failed fast retrains.

5.12.4.3.3 adslAtucIntervalSesL

For the each interval, this object reports the number of seconds during which there have been severely errored seconds-line.

5.12.4.3.4 adslAtucIntervalUasL

For the each interval, this object reports the number of seconds during which there have been unavailable seconds-line.

5.12.4.4 adslAturPerfDataExtTable

This table contains ADSL physical line counters not defined in the adslAturPerfDataTable (Section 5.11.2.7, Page 185) from the ADSL-LINE-MIB [RFC 2662]. Each entry corresponds to an ADSL line.

Augments:

- adslAturPerfDataTable (Section 5.11.2.7, Page 185)

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- adslAturPerfDataExtTable - { adslExtMibObjects 20 } 1.3.6.1.2.1.10.94.3.1.20
- adslAturPerfDataExtEntry - { adslAturPerfDataExtTable 1 } 1.3.6.1.2.1.10.94.3.1.20.1

Table 119. adslAturPerfDataExtTable

Object	OID	Syntax	Access	Status	Supported
adslAturPerfStatSesL	{ adslAtucIntervalExtEntry 1 }	Counter32	read-only	current	Y
adslAturPerfStatUasL	{ adslAtucIntervalExtEntry 2 }	Counter32	read-only	current	Y
adslAturPerfCurr15MinSesL	{ adslAtucIntervalExtEntry 3 }	PerfCurrentCount	read-only	current	Y
adslAturPerfCurr15MinUasL	{ adslAtucIntervalExtEntry 4 }	PerfCurrentCount	read-only	current	Y
adslAturPerfCurr1DaySesL	{ adslAtucIntervalExtEntry 5 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturPerfCurr1DayUasL	{ adslAtucIntervalExtEntry 6 }	AdslPerfCurrDayCount	read-only	current	Y
adslAturPerfPrev1DaySesL	{ adslAtucIntervalExtEntry 7 }	AdslPerfPrevDayCount	read-only	current	Y
adslAturPerfPrev1DayUasL	{ adslAtucIntervalExtEntry 8 }	AdslPerfPrevDayCount	read-only	current	Y

5.12.4.4.1 adslAturPerfStatSesL

The value of this object reports the count of severely errored second-line since the last agent reset.

5.12.4.4.2 adslAturPerfStatUasL

The value of this object reports the count of unavailable seconds-line since the last agent reset.

5.12.4.4.3 adslAturPerfCurr15MinSesL

For the current 15-minute interval, this object reports the current number of seconds during which there have been severely errored seconds-line.

5.12.4.4.4 adslAturPerfCurr15MinUasL

For the current 15-minute interval, this object reports the current number of seconds during which there have been available seconds-line.

5.12.4.4.5 adslAturPerfCurr1DaySesL

For the current day as measured by adslAturPerfCurr1DayTimeElapsed (Section 5.11.2.7.12, Page 187) [RFC 2662], adslAturPerfCurr1DaySesL reports the number of seconds during which there have been severely errored seconds-line.

5.12.4.4.6 adslAturPerfCurr1DayUasL

For the current day as measured by adslAturPerfCurr1DayTimeElapsed (Section 5.11.2.7.12, Page 187) [RFC 2662], this object reports the number of seconds during which there have been unavailable seconds-line.

5.12.4.4.7 adslAturPerfPrev1DaySesL

For the previous day, this object reports the number of seconds during which there were severely errored seconds-line.

5.12.4.4.8 adslAturPerfPrev1DayUasL

For the previous day, this object reports the number of seconds during which there were severely errored seconds-line.

5.12.4.5 adslAturIntervalExtTable

This table provides one row for each ATU-R performance data collection interval for ADSL physical interfaces whose IfEntries' ifType (Section 5.3.4.3.3, Page 41) is equal to adsl(94).

Augments:

- adslAturIntervalTable (Section 5.11.2.9, Page 189)

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- adslAturIntervalNumber (Section 5.11.2.9.1, Page 190)

OIDs:

- adslAturIntervalExtTable - { adslExtMibObjects 21 } 1.3.6.1.2.1.10.94.3.1.21
- adslAturIntervalExtEntry - { adslAturIntervalExtTable 1 } 1.3.6.1.2.1.10.94.3.1.21.1

Table 120. adslAturIntervalExtTable

Object	OID	Syntax	Access	Status	Supported
adslAturIntervalSesL	{ adslAturIntervalExtEntry 1 }	PerfIntervalCount	read-only	current	Y
adslAturIntervalUasL	{ adslAturIntervalExtEntry 2 }	PerfIntervalCount	read-only	current	Y

5.12.4.5.1 adslAturIntervalSesL

For the each interval, this object reports the number of seconds during which there have been severely errored seconds-line.

5.12.4.5.2 adslAturIntervalUasL

For the each interval, this object reports the number of seconds during which there have been unavailable seconds-line.

5.12.4.6 adslConfProfileExtTable

This table extends the ADSL line profile configuration information in the adslLineConfProfileTable (Section 5.11.2.14, Page 198) from the ADSL-LINE-MIB [RFC 2662] by adding the ability to configure the ADSL physical line mode. Each entry corresponds to an ADSL line profile.

Augments:

- adslLineConfProfileTable (Section 5.11.2.14, Page 198)

Index:

- adslLineConfProfileName (Section 5.11.2.14.1, Page 200)

OIDs:

- adslConfProfileExtTable - { adslExtMibObjects 22 } 1.3.6.1.2.1.10.94.3.1.22
- adslConfProfileExtEntry - { adslConfProfileExtTable 1 } 1.3.6.1.2.1.10.94.3.1.22.1

Table 121. adslConfProfileExtTable

Object	OID	Syntax	Access	Status	Supported
adslConfProfileLineType	{ adslConfProfileExtEntry 1 }	INTEGER noChannel (1) fastOnly (2) interleavedOnly (3) fastOrInterleaved (4) fastAndInterleaved (5)	read-create	current	Y

5.12.4.6.1 adslConfProfileLineType

This object is used to configure the ADSL physical line mode and can take on the following values:

- noChannel (1) - no channels exist
- fastOnly (2) - fast channel exists only
- interleavedOnly (3) - interleaved channel exists only
- fastOrInterleaved (4) - either fast or interleaved channels can exist, but only one at any time
- fastAndInterleaved (5) - both fast or interleaved channels exist

In the case when no value has been set, the MIB recommended default value is noChannel(1).

Valid Object Value(s)

- fastOnly (2)
- interleavedOnly (3)

5.12.4.7 adslAlarmConfProfileExtTable

This table extends the adslLineAlarmConfProfileTable (Section 5.11.2.15, Page 204) and provides threshold parameters for all the counters defined in this MIB module. Each entry corresponds to an ADSL alarm profile.

Augments:

- adslLineAlarmConfProfileTable (Section 5.11.2.15, Page 204)

Index:

- adslLineAlarmConfProfileName (Section 5.11.2.15.1, Page 206)

OIDs:

- adslAlarmConfProfileExtTable - { adslExtMibObjects 23 } 1.3.6.1.2.1.10.94.3.1.23
- adslAlarmConfProfileExtEntry - { adslAlarmConfProfileExtTable 1 } 1.3.6.1.2.1.10.94.3.1.23.1

Table 122. adslAlarmConfProfileExtTable

Object	OID	Syntax	Access	Status	Sup-ported
adslAtucThreshold15MinFailedFastR	{ adslAlarmConfProfileExtEntry 1 }	Integer32 (0…900)	read-create	current	Y
adslAtucThreshold15MinSesL	{ adslAlarmConfProfileExtEntry 2 }	Ineger32 (0…900)	read-create	current	Y
adslAtucThreshold15MinUasL	{ adslAlarmConfProfileExtEntry 3 }	Ineger32 (0…900)	read-create	current	Y
adslAturThreshold15MinSesL	{ adslAlarmConfProfileExtEntry 4 }	Integer32 (0…900)	read-create	current	Y
adslAturThreshold15MinUasL	{ adslAlarmConfProfileExtEntry 5 }	Integer32 (0…900)	read-create	current	Y

5.12.4.7.1 adslAtucThreshold15MinFailedFastR

The first time the value of the corresponding instance of adslAtucPerfCurr15MinFailedFastR (Section 5.12.4.2.2, Page 215) reaches or exceeds this value within a given 15-minute performance data collection period, an adslAtucFailedFastRThreshTrap (Section 5.12.5.1.1, Page 222) notification is generated. The value “0” will disable the notification. The recommended MIB default value of this object is “0.”

5.12.4.7.2 adslAtucThreshold15MinSesL

The first time the value of the corresponding instance of adslAtucPerfCurr15MinSesL (Section 5.12.4.2.7, Page 215) reaches or exceeds this value within a given 15-minute performance data collection period, an adslAtucSesLThreshTrap (Section 5.12.5.1.2, Page 222) notification is generated. The value “0” will disable the notification. The MIB recommended default value of this object is “0”.

Note: the RFC used adslAtucPerf15MinSesL. This seems to be a typo as that object does not exist.

5.12.4.7.3 adslAtucThreshold15MinUasL

The first time the value of the corresponding instance of adslAtucPerfCurr15MinUasL (Section 5.12.4.2.8, Page 215) reaches or exceeds this value within a given 15-minute performance data collection period, an adslAtucUasLThreshTrap (Section 5.12.5.1.3, Page 222) notification is generated. The value “0” will disable the notification. The MIB recommended default value of this object is “0”.

Note: the RFC used adslAtucPerf15MinUasL. This seems to be a typo as that object does not exist.

5.12.4.7.4 adslAturThreshold15MinSesL

The first time the value of the corresponding instance of adslAturPerfCurr15MinSesL (Section 5.12.4.4.3, Page 217) reaches or exceeds this value within a given 15-minute performance data collection period, an adslAturSesLThreshTrap (Section 5.12.5.2.1, Page 223) notification is generated. The value “0” will disable the notification. The MIB recommended default value of this object is “0.”

Units are in seconds.

Note: the RFC used adslAturPerfCurr15MinSesL. This seems to be a typo as that object does not exist.

5.12.4.7.5 adslAturThreshold15MinUasL

The first time the value of the corresponding instance of adslAturPerfCurr15MinUasL (Section 5.12.4.4.4, Page 217) reaches or exceeds this value within a given 15-minute performance data collection period, an adslAturUasLThreshTrap (Section 5.12.5.2.2, Page 223) notification is generated. The value “0” will disable the notification. The MIB recommended default value of this object is 0."

Units are in seconds.

Note: the RFC used adslAturPerfCurr15MinUasL. This seems to be a typo as that object does not exist.

5.12.5 Traps

This section defines the traps generated.

5.12.5.1 adslExtAtucTraps

OIDs:

- adslExtTraps - { adslExtMibObjects 24 } 1.3.6.1.2.1.10.94.3.1.24
- adslExtAtucTraps - { adslExtTraps 1 } 1.3.6.1.2.1.10.94.3.1.24.1
- adslExtAtucTapsPrefix - { adslExtAtucTraps 0 } 1.3.6.1.2.1.10.94.3.1.24.1.0

Table 123. ADSL Line Extension MIB Atuc Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
adslAtucFailedFastRThreshTrap	{ adslExtAtucTrasPrefix 1 }	adslAtucPerfCurr15MinFailedFastR adslAtucThreshold15MinFailedFastR	current	N
adslAtucSesLThreshTrap	{ adslExtAtucTrapsPrefix 2 }	adslAtucPerfCurr15MinSesL adslAtucThreshold15MinSesL	current	Y
adslAtucUasLThreshTrap	{ adslExtAtucTrapsPrefix 3 }	adslAtucPerfCurr15MinUasL adslAtucThreshold15MinUasL	current	Y

5.12.5.1.1 adslAtucFailedFastRThreshTrap

Failed Fast Retrains 15-minute threshold reached.

5.12.5.1.2 adslAtucSesLThreshTrap

Severely errored seconds-line 15-minute threshold reached.

5.12.5.1.3 aslAtucUasLThreshTrap

Unavailable seconds-line 15-minute threshold reached.

5.12.5.2 adslExtAtruTraps

OIDs:

- adslExtTraps - { adslExtMibObjects 24 } 1.3.6.1.2.1.10.94.3.1.24
- adslExtAturTraps - { adslExtTraps 2 } 1.3.6.1.2.1.10.94.3.1.24.2
- adslExtAturTrapsPrefix - { adslExtAturTraps 0 } 1.3.6.1.2.1.10.94.3.1.24.2.0

Table 124. ADSL Line Extension MIB Atur Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
adslAturSesLThreshTrap	{ adslExtAturTrapsPrefix 1 }	adslAturPerfCurr15MinSesL adslAturThreshold15MinSesL	current	Y
adslAturUasLThreshTrap	{ adslExtAturTrapsPrefix 2 }	adslAturPerfCurr15MinUasL adslAturThreshold15MinUasL	current	Y

5.12.5.2.1 adslAturSesLThreshTrap

Severely errored seconds-line 15-minute threshold reached.

5.12.5.2.2 adslAturUasLThreshTrap

Unavailable seconds-line 15-minute threshold reached.

5.12.6 Conformance Information

5.12.6.1 Compliance

5.12.6.1.1 adslExtLineMibAtucCompliance

The compliance statement for SNMP entities which represent ADSL ATU-C interfaces.

The following groups are mandatory groups:

- adslExtLineGroup
- adslExtLineConfProfileControlGroup
- adslExtLineAlarmConfProfileGroup

The following are optional groups:

- adslExtAtucPhysPerfCounterGroup - This group is optional. Implementations which require continuous ATU-C physical event counters should implement this group.
- adslExtAturPhysPerfCounterGroup - This group is optional. Implementations which require continuous ATU-R physical event counters should implement this group.
- adslExtNotificationsGroup - This group is optional. Implementations which support TCA (Threshold Crossing Alert) should implement this group.

The following objects have a MIN-ACCESS defined:

- adslAtucThreshold15MinFailedFastR - MIN-ACCESS read-write
  Read-write access is applicable only when static profiles as defined in ADSL Line MIB [RFC 2662] are implemented.
- adslAtucThreshold15MinSesL - MIN-ACCESS read-write
  Read-write access is applicable only when static profiles as defined in ADSL Line MIB [RFC 2662] are implemented.
- adslAtucThreshold15MinUasL - MIN-ACCESS read-write
  Read-write access is applicable only when static profiles as defined in ADSL Line MIB [RFC 2662] are implemented.
- adslAturThreshold15MinSesL - MIN-ACCESS read-write
  Read-write access is applicable only when static profiles as defined in ADSL Line MIB [RFC 2662] are implemented.
- adslAturThreshold15MinUasL - MIN-ACCESS read-write
  Read-write access is applicable only when static profiles as defined in ADSL Line MIB [RFC 2662] are implemented.
- adslLineConfProfileDualLite - MIN-ACCESS read-only
  Read-only access is applicable only when static profiles as defined in ADSL Line MIB [RFC 2662] are implemented.

5.12.6.2 Conformance

Table 125. ADSL-LINE-EXT-MIB Conformance (1 of 2)

Group	Description	Objects	Supported
adslExtLineGroup	A collection of objects providing extended configuration information about an ADSL Line.	adslLineConfProfileDualLite	N
		adslLineTransAutcCap	Y
		adslLineTransAtucConfig	Y
		adslLineTransAtucActual	Y
		adslLineGlitePowerState	Y
adslExtAtucPhysPerfCounterGroup	A collection of objects providing raw performance counts on an ADSL Line (ATU-C end).	adslAtucPerfStatFastR	Y
		adslAtucPerfStatFailedFastR	Y
		adslAtucPerfCurr15MinFastR	Y
		adslAtucPerfCurr15MinFailedFastR	Y
		adslAtucPerfCurr1DayFastR	Y
		adslAtucPerfCurr1DayFailedFastR	Y
		adslAtucPerfPrev1DayFastR	Y
		adslAtucPerfPrev1DayFailedFastR	Y
		adslAtucPerfStatSesL	Y
		adslAtucPerfStatUasL	Y
		adslAtucPerfCurr15MinSesL	Y
		adslAtucPerfCurr15MinUasL	Y
		adslAtucPerfCurr1DaySesL	Y
		adslAtucPerfCurr1DayUasL	Y
		adslAtucPerfPrev1DaySesL	Y
		adslAtucPerfPrev1DayUasL	Y
		adslAtucIntervalFastR	Y
		adslAtucIntervalFailedFastR	Y
		adslAtucIntervalSesL	Y
		adslAtucIntervalUasL	Y

Table 125. ADSL-LINE-EXT-MIB Conformance (2 of 2)

Group	Description	Objects	Supported
adslExtAturPhysPerfCounterGroup	A collection of objects providing raw performance counts on an ADSL Line (ATU-C end).	adslAturPerfStatSesL	Y
		adslAturPerfStatUasL	Y
		adslAturPerfCurr15MinSesL	Y
		adslAturPerfCurr15MinUasL	Y
		adslAturPerfCurr1DaySesL	Y
		adslAturPerfCurr1DayUasL	Y
		adslAturPerfPrev1DaySesL	Y
		adslAturPerfPrev1DayUasL	Y
		adslAturIntervalSesL	Y
		adslAturIntervalUasL	Y
adslExtLineConfProfileControlGroup	A collection of objects providing profile control for the ADSL system.	adslConfProfileLineType	Y
adslExtLineAlarmConfProfileGroup	A collection of objects providing alarm profile control for the ADSL system.	adslAutcThreshold15MinFailedFastR	Y
		adslAtucThreshold15MinSesL	Y
		adslAtucThreshold15MinUasL	Y
		adslAturThreshold15MinSesL	Y
		adslAturThreshold15MinUasL	Y
adslExtNotificationsGroup	The collection of ADSL extension notifications.	adslAutcFailedFastRThreshTrap	N
		adslAutcSesLThreshTrap	Y
		adslAutcUasLThreshTrap	Y
		adslAutrSesLThreshTrap	Y
		adslAutrUasLThreshTrap	Y

5.13 ATM-MIB — RFC 2515

5.13.1 Introduction

This MIB describes objects for managing ATM-based interfaces. The RFC used as a reference was dated February 1999. The LAST-UPDATED for the atmMIB was “9810191200Z.”

5.13.2 OIDs

• iso	- { 1 }	1
• org	- { iso 3 }	1.3
• dod	- { org 6 }	1.3.6
• internet	- { dod 1 }	1.3.6.1
• mgmt	- { internet 2 }	1.3.6.1.2
• mib-2	- { mgmt 1 }	1.3.6.1.2.1
• transmission	- { mib-2 10 }	1.3.6.1.2.1.10
• atmMIB	- { mib-2 37 }	1.3.6.1.2.1.10.37
• atmMIBObjects	- { atmMIB 1 }	1.3.6.1.2.1.10.37.1
• atmTrafficDescriptorTypes	- { atmMIBObjects 1 }[1]	1.3.6.1.2.1.10.37.1.1
• atmInterfaceConfTable	- { atmMIBObjects 2 }	1.3.6.1.2.1.10.37.1.2
• atmInterfaceDs3PlcpTable	- { atmMIBObjects 3 }	1.3.6.1.2.1.10.37.1.3
• atmInterfaceTCTable	- { atmMIBObjects 4 }	1.3.6.1.2.1.10.37.1.4
• atmTrafficDescrParamTable	- { atmMIBObjects 5 }	1.3.6.1.2.1.10.37.1.5
• atmVplTable	- { atmMIBObjects 6 }	1.3.6.1.2.1.10.37.1.6
• atmVclTable	- { atmMIBObjects 7 }	1.3.6.1.2.1.10.37.1.7
• atmVpCrossConnectIndexNext	- { atmMIBObjects 8 }	1.3.6.1.2.1.10.37.1.8
• atmVpCrossConnectTable	- { atmMIBObjects 9 }	1.3.6.1.2.1.10.37.1.9
• atmVcCrossconnectIndexNext	- { atmMIBObjects 10 }	1.3.6.1.2.1.10.37.1.10
• atmVcCrossConnectTable	- { atmMIBObjects 11 }	1.3.6.1.2.1.10.37.1.11
• aal5VccTable	- { atmMIBObjects 12 }	1.3.6.1.2.1.10.37.1.12
• atmTrafficDescrParamIndexNext	- { atmMIBObjects 13 }	1.3.6.1.2.1.10.37.1.13

5.13.3 Textual Conventions

The following textual conventions were cited in RFC 2514, *ATM TCs and OBJECT-IDENTITIES*, February 1999:

- AtmAddr - An ATM address. The semantics are implied by the length.
  — SYNTAX OCTET STRING (SIZE(0…40)
    - no address (0 octets)
    - E.164 (8 octets)
    - NSAP (20 octets)

  In addition, when subaddresses are used the AtmAddr may represent the concatenation of address and subaddress. The associated address types are:

    - E.164, E.164 (16 octets)
    - E.164, NSAP (28 octets)
    - NSAP, NSAP (40 octets)

  Address lengths other than defined in this definition imply address types defined elsewhere. Note: The E.164 address is encoded in BCD format.

1. From RFC 2514, *ATM TCs and OBJECT-IDENTITIES*, February, 1999.

- AtmConnCastType - The type of topology of a connection (point- to-point, point-to-multipoint). In the case of point-to-multipoint, the orientation of this VPL or VCL in the connection. On a host:
  - •p2mpRoot indicates that the host is the root of the p2mp connection.
  - •p2mpLeaf indicates that the host is a leaf of the p2mp connection.

  On a switch interface:
  - •p2mpRoot indicates that cells received by the switching fabric from the interface are from the root of the p2mp connection.
  - •p2mpLeaf indicates that cells transmitted to the interface from the switching fabric are to the leaf of the p2mp connection.

  — SYNTAX INTEGER
  - p2p(1)
  - p2mpRoot(2)
  - p2mpLeav(3)

- AtmConnKind - The type of call control used for an ATM connection at a particular interface. The use is as follows:
  - •pvc(1)- Virtual link of a PVC. Should not be used for an PVC/SVC (that is, Soft PVC) crossconnect.
  - •svcIncoming(2)- Virtual link established after a received signaling request to setup an SVC.
  - •svcOutgoing(3)- Virtual link established after a transmitted or forwarded signaling request to setup an SVC.
  - •spvcInitiator(4)- Virtual link at the PVC side of an SVC/PVC crossconnect, where the switch is the initiator of the Soft PVC setup.
  - •spvcTarget(5)- Virtual link at the PVC side of an SVC/PVC crossconnect, where the switch is the target of the Soft PVC setup.

  For PVCs, a pvc virtual link is always cross-connected to a pvc virtual link.

  For SVCs, an svcIncoming virtual link is always cross-connected to an svcOutgoing virtual link.

  For Soft PVCs, an spvcInitiator is either cross-connected to an svcOutgoing or an spvcTarget, and an spvcTarget is either cross-connected to an svcIncoming or an spvcInitiator.

SYNTAX INTEGER

- pvc(1)
- svcIncoming(2)
- svcOutgoing(3)
- spvcInitiator(4)
- spvcTarget(5)

- AtmIlmiNetworkPrefix - A network prefix used for ILMI address registration. In the case of ATM endsystem addresses (AESAs), the network prefix is the first 13 octets of the address which includes the AFI, IDI, and HO-DSP fields. In the case of native E.164 addresses, the network prefix is the entire E.164 address encoded in 8 octets, as if it were an E.164 IDP in an ATM endsystem address structure.

  Syntax: OCTET STRING (SIZE(8|13))

- AtmInterfaceType - The connection setup procedures used for the identified interface.
  - •Other
  - •Connection setup procedures other than hose listed below.
  - •Auto-configuration
  - •Indicates that the connection setup procedures are to be determined dynamically, or that determination has not yet been completed. One such mechanism is via ATM Forum ILMI auto-configuration procedures.
  - •ITU-T DSS2:
    - —ITU-T Recommendation Q.2931, Broadband Integrated Service Digital Network (B-ISDN) Digital Subscriber Signalling System No.2 (DSS2) User-Network Interface (UNI) Layer 3 Specification for Basic Call/Connection Control (September 1994)
    - —ITU-T Draft Recommendation Q.2961, B-ISDN DSS 2 Support of Additional Traffic Parameters (May 1995)
    - —ITU-T Draft Recommendation Q.2971, B-ISDN DSS 2 User Network Interface Layer 3 Specification for Point-to-multipoint Call/connection Control (May 1995)
  - •ATM Forum UNI 3.0
    ATM Forum, ATM User-Network Interface, Version 3.0 (UNI 3.0) Specification, (1994).
  - •ATM Forum UNI 3.1
    ATM Forum, ATM User-Network Interface, Version 3.1 (UNI 3.1) Specification, (November 1994).
  - •ATM Forum UNI Signalling 4.0
    ATM Forum, ATM User-Network Interface (UNI) Signalling Specification Version 4.0, af-sig-0061.000 (June 1996).
  - •ATM Forum IISP (based on UNI 3.0 or UNI 3.1)
    Interim Inter-switch Signaling Protocol (IISP) Specification, Version 1.0, af-pnni-0026.000, (December 1994).
  - •ATM Forum PNNI 1.0
    ATM Forum, Private Network-Network Interface Specification, Version 1.0, af-pnni-0055.000, (March 1996).
  - •ATM Forum B-ICI
    ATM Forum, B-ICI Specification, Version 2.0, af-bici-0013.002, (November 1995).
  - •ATM Forum UNI PVC Only
    An ATM Forum compliant UNI with the signalling disabled.
  - •ATM Forum NNI PVC Only
    An ATM Forum compliant NNI with the signalling disabled.

  — SYNTAX INTEGER
    - other(1)
    - autoConfig(2)
    - ituDss2(3)
    - atmfUni3Dot0(4)
    - atmfUni3Dot1(5)
    - atmfUni4Dot0(6)
    - atmfIispUni3Dot0(7)
    - atmfIispUni3Dot1(8)
    - atmfIispUni4Dot0(9)

- atmfPnni1Dot0(10)
- atmfBici2Dot0(11)
- atmfUniPvcOnly(12)
- atmfNniPvcOnly(13)

- AtmServiceCategory - The service category for a connection.
  — SYNTAX INTEGER
  - other(1) - none of the following
  - cbr(2) - constant bit rate
  - rtVbr(3) - real-time variable bit rate
  - nrtVbr(4) - non real-time variable bit rate
  - abr(5) - available bit rate
  - ubr(6) - unspecified bit rate

- AtmSigDescrParamIndex - The value of this object identifies a row in the atmSigDescrParamTable. The value 0 signifies that none of the signalling parameters defined in the atmSigDescrParamTable are applicable.
  — SYNTAX INTEGER (0…2147483647)

- AtmTrafficDescrParamIndex - The value of this object identifies a row in the atmTrafficDescrParamTable. The value 0 signifies that no row has been identified.
  — SYNTAX INTEGER (0…2147483647)

- AtmVcIdentifier - The VCI value for a VCL. The maximum VCI value cannot exceed the value allowable by atmInterfaceMaxVciBits defined in ATM-MIB.

  SYNTAX INTEGER (0…65535)

- AtmVpIdentifier - The VPI value for a VPL or VCL. The value VPI=0 is only allowed for a VCL. For ATM UNIs supporting VPCs the VPI value ranges from 0 to 255. The VPI value 0 is supported for ATM UNIs conforming to the ATM Forum UNI 4.0 Annex 8 (Virtual UNIs) specification. For ATM UNIs supporting VCCs the VPI value ranges from 0 to 255. For ATM NNIs the VPI value ranges from 0 to 4095. The maximum VPI value cannot exceed the value allowable by atmInterfaceMaxVpiBits defined in ATM-MIB.

  SYNTAX INTEGER (0…4095)

- AtmVorXAdminStatus - The value determines the desired administrative status of a virtual link or cross-connect. The up and down states indicate that the traffic flow is enabled or disabled respectively on the virtual link or cross-connect.

  SYNTAX INTEGER

  - up(1)
  - down(2)

- AtmVorXLastChange - The value of MIB II's sysUpTime at the time a virtual link or cross-connect entered its current operational state. If the current state was entered prior to the last re-initialization of the agent then this object contains a zero value.
  — SYNTAX TimeTicks (Section 4.11, Page 15)

- AtmVorXOperStatus - The value determines the operational status of a virtual link or cross-connect. The up and down states indicate that the traffic flow is enabled or disabled respectively on the virtual link or cross-connect. The unknown state indicates that the state of it cannot be determined. The state is down or unknown if the supporting ATM interface(s) is down or unknown respectively.

  SYNTAX INTEGER

- up(1)
- down(2)
- unknown(3)

5.13.4 MIB Objects

The following tables are supported in this MIB:

- ATM Interface configuration table
- ATM Interface DS3 PLCP table
- ATM Interface TC Sublayer table
- ATM Traffic Descriptor table
- ATM Interface VPL configuration table
- ATM Interface VCL configuration table
- ATM VP Cross connect table (for PVCs)
- ATM VC Cross connect table (for PVCs)
- ATM Interface AAL5 VCC performance statistics table

Table 126. ATM-MIB Objects

Object	OID	Syntax	Access	Status	Supported
atmInterfaceConfTable	{ atmMIBObjects 2 }	Sequence of atmInterfaceConfEntry	not-accessible	current	N
atmInterfaceDs3PlcpTable	{ atmMIBObjects 3 }	Sequence of atmInterfaceDs3PlcpEntry	not-accessible	current	N
atmInterfaceTCTable	{ atmMIBObjects 4 }	Sequence of atmInterfaceTCEntry	not-accessible	current	Y
atmTrafficDescrParamTable	{ atmMIBObjects 5 }	Sequence of atmTrafficDescrParamEntry	not-accessible	current	N
atmVplTable	{ atmMIBObjects 6 }	Sequence of atmVplEntry	not-accessible	current	N
atmVclTable	{ atmMIBObjects 7 }	Sequence of atmVclEntry	not-accessible	current	Y
atmVpCrossConnectIndexNext	{ atmMIBObjects 8 }	INTEGER(0…2147483647)	read-only	current	N
atmVpCrossConnectTable	{ atmMIBObjects 9 }	Sequence of atmVpCrossConnectEntry	not-accessible	current	N
atmVcCrossConnectIndexNext	{ atmMIBObjects 10 }	INTEGER(0…2147483647)	read-only	current	N
atmVcCrossConnectTable	{ atmMIBObjects 11 }	Sequence of atmVcCrossConnectEntry	not-accessible	current	N
aal5VccTable	{ atmMIBObjects 12 }	Sequence of aal5VccTable	not-accessible	current	Y
atmTrafficDescrParamIndexNext	{ atmMIBObjects 13 }	INTEGER(0…2147483647)	not-accessible	current	N

5.13.4.1 atmInterfaceConfTable

The ATM Interface Configuration Parameters Table contains ATM specific configuration information associated with an ATM interface beyond those supported using the ifTable. There is one entry per ATM interface port. This table is a sequence of atmInterfaceConfEntry.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- atmInterfaceConfTable - { atmMIBObjects 2 } 1.3.6.1.2.1.10.37.1.2
- atmInterfaceConfEntry - { atmInterfaceConfTable 1} 1.3.6.1.2.1.10.37.1.2.1

Table 127. atmInterfaceConfTable

Object	OID	Syntax	Access	Status	Supported
atmInterfaceMaxVpcs	{ atmInterfaceConfEntry 1 }	INTEGER (0…4096)	read-write	current	N
atmInterfaceMaxVccs	{ atmInterfaceConfEntry 2 }	INTEGER (0…65536)	read-write	current	N
atmInterfaceConfVpcs	{ atmInterfaceConfEntry 3 }	INTEGER (0…4096)	read-only	current	N
atmInterfaceConfVccs	{ atmInterfaceConfEntry 4 }	INTEGER (0…65536)	read-only	current	N
atmInterfaceMaxActiveVpiBits	{ atmInterfaceConfEntry 5 }	INTEGER (0…12)	read-write	current	N
atmInterfaceMaxActiveVciBits	{ atmInterfaceConfEntry 6 }	INTEGER (0…16)	read-write	current	N
atmInterfaceIlmiVpi	{ atmInterfaceConfEntry 7 }	AtmVpIdentifier	read-write	current	N
atmInterfaceIlmiVci	{ atmInterfaceConfEntry 8 }	AtmVpIdentifier	read-write	current	N
atmInterfaceAddressType	{ atmInterfaceConfEntry 9 }	INTEGER private(1) nsapE164(2) nativeE164(3) other(4)	read-only	deprecated	N
atmInterfaceAdminAddress	{ atmInterfaceConfEntry 10 }	AtmAddr	read-only	deprecated	N
atmInterfaceMyNeighborIpAddress	{ atmInterfaceConfEntry 11 }	IpAddress	read-write	current	N
atmInterfaceMyNeighborIfName	{ atmInterfaceConfEntry 12 }	DisplayString	read-write	current	N
atmInterfaceCurrentMaxVpiBits	{ atmInterfaceConfEntry 13 }	INTEGER (0…12)	read-only	current	N
atmInterfaceCurrentMaxVciBits	{ atmInterfaceConfEntry 14 }	INTEGER (0…16)	read-only	current	N
atmInterfaceSubscrAddress	{ atmInterfaceConfEntry 15 }	AtmAddr	read-write	current	N

5.13.4.1.1 atmInterfaceMaxVpcs

The maximum number of VPCs (PVPCs and SVPCs) supported at this ATM interface. At the ATM UNI, the maximum number of VPCs (PVPCs and SVPCs) ranges from 0 to 256 only.

5.13.4.1.2 atmInterfaceMaxVccs

The maximum number of VCCs (PVCCs and SVCCs) supported at this ATM interface.

5.13.4.1.3 atmInterfaceConfVpcs

The number of VPCs (PVPC, Soft PVPC and SVPC) currently in use at this ATM interface. It includes the number of PVPCs and Soft PVPCs that are configured at the interface, plus the number of SVPCs that are currently established at the interface.

At the ATM UNI, the configured number of VPCs (PVPCs and SVPCs) can range from 0 to 256 only.

5.13.4.1.4 atmInterfaceConfVccs

The number of VCCs (PVCC, Soft PVCC and SVCC) currently in use at this ATM interface. It includes the number of PVCCs and Soft PVCCs that are configured at the interface, plus the number of SVCCs that are currently established at the interface.

5.13.4.1.5 atmInterfaceMaxActiveVpiBits

The maximum number of active VPI bits configured for use at the ATM interface. At the ATM UNI, the maximum number of active VPI bits configured for use ranges from 0 to 8 only.

5.13.4.1.6 atmInterfaceMaxActiveVciBits

The maximum number of active VCI bits configured for use at this ATM interface.

5.13.4.1.7 atmInterfaceIlmiVpi

The VPI value of the VCC supporting the ILMI at this ATM interface. If the values of atmInterfaceIlmiVpi and atmInterfaceIlmiVci are both equal to zero then the ILMI is not supported at this ATM interface.

The MIB recommended default value for this object is { 0 }.

5.13.4.1.8 atmInterfaceIlmiVci

The VCI value of the VCC supporting the ILMI at this ATM interface. If the values of atmInterfaceIlmiVpi and atmInterfaceIlmiVci are both equal to zero then the ILMI is not supported at this ATM interface.

The MIB recommended default value for this object is { 16 }.

5.13.4.1.9 atmInterfaceAddressType

The type of primary ATM address configured for use at this ATM interface.

- private(1),
- nsapE164(2),
- nativeE164(3),
- other(4)

5.13.4.1.10 atmInterfaceAdminAddress

The primary address assigned for administrative purposes, for example, an address associated with the service provider side of a public network UNI (thus, the value of this address corresponds with the value of ifPhysAddress at the host side). If this interface has no assigned administrative address, or when the address used for administrative purposes is the same as that used for ifPhysAddress, then this is an octet string of zero length.

Note: The atmInterfaceAdminAddress object has been replaced by atmInterfaceSubscrAddress.

5.13.4.1.11 atmInterfaceMyNeighborIpAddress

The IP address of the neighbor system connected to the far end of this interface, to which a Network Management Station can send SNMP messages, as IP datagrams sent to UDP port 161, in order to access network management information concerning the operation of that system. Note that the value of this object may be obtained in different ways, for example, by manual configuration, or through ILMI interaction with the neighbor system.

5.13.4.1.12 atmInterfaceMyNeighborIfName

The textual name of the interface on the neighbor system on the far end of this interface, and to which this interface connects. If the neighbor system is manageable through SNMP and supports the object ifName, the value of this object must be identical with that of ifName for the ifEntry of the lowest level physical interface for this port. If this interface does not have a textual name, the value of this object is a zero length string. Note that the value of this object may be obtained in different ways, for example, by manual configuration, or through ILMI interaction with the neighbor system.

5.13.4.1.13 atmInterfaceCurrentMaxVpiBits

The maximum number of VPI Bits that may currently be used at this ATM interface. The value is the minimum of atmInterfaceMaxActiveVpiBits, and the atmInterfaceMaxActiveVpiBits of the interface's UNI/NNI peer.

If the interface does not negotiate with its peer to determine the number of VPI Bits that can be used on the interface, then the value of this object must equal atmInterfaceMaxActiveVpiBits.

5.13.4.1.14 atmInterfaceCurrentMaxVciBits

The maximum number of VCI Bits that may currently be used at this ATM interface. The value is the minimum of atmInterfaceMaxActiveVciBits, and the atmInterfaceMaxActiveVciBits of the interface's UNI/NNI peer.

If the interface does not negotiate with its peer to determine the number of VCI Bits that can be used on the interface, then the value of this object must equal atmInterfaceMaxActiveVciBits.

5.13.4.1.15 atmInterfaceSubscrAddress

The identifier assigned by a service provider to the network side of a public network UNI. If this interface has no assigned service provider address, or for other interfaces this is an octet string of zero length.

5.13.4.2 atmInterfaceDs3PlcpTable

The ATM Interface DS3 PLCP Table contains ATM interface DS3 PLCP parameters and state variables, one entry per ATM interface port. This table is a sequence of atmInterfaceDs3PlcpEntry which contains DS3 PLCP parameters and state variables at the ATM interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- atmInterfaceDs3PlcpTable - { atmMIBObjects 3 } 1.3.6.1.2.1.10.37.1.3
- atmInterfaceDs3PlcpEntry - { atmInterfaceDs3PlcpTable 1} 1.3.6.1.2.1.10.37.1.3.1

Table 128. atmInterfaceDs3PlcpTable

Object	OID	Syntax	Access	Status	Supported
atmInterfaceDs3PlcpSEFSs	{ atmInterfaceDs3PlcpEntry 1 }	Counter32	read-only	current	N
atmInterfaceDs3PlcpAlarmState	{ atmInterfaceDs3PlcpEntry 2 }	INTEGER noAlarm(1) receivedFarEndAlarm(2) incomingLOF(3)	read-only	current	N
atmInterfaceDs3PlcpUASs	{ atmInterfaceDs3PlcpEntry 3 }	Counter32	read-only	current	N

5.13.4.2.1 atmInterfaceDs3PlcpSEFSs

The number of DS3 PLCP Severely Errored Framing Seconds (SEFS). Each SEFS represents a one-second interval which contains one or more SEF events.

5.13.4.2.2 atmInterfaceDs3PlcpAlarmState

This variable indicates if there is an alarm present for the DS3 PLCP. The value receivedFarEndAlarm means that the DS3 PLCP has received an incoming Yellow Signal, the value incomingLOF means that the DS3 PLCP has declared a loss of frame (LOF) failure condition, and the value noAlarm means that there are no alarms present. Transition from the failure to the no alarm state occurs when no defects (for example, LOF) are received for more than 10 seconds.

- noAlarm(1)
- receivedFarEndAlarm(2)
- incomingLOF(3)

5.13.4.2.3 atmInterfaceDs3PlcpUASs

The counter associated with the number of Unavailable Seconds encountered by the PLCP.

5.13.4.3 atmInterfaceTCTable

The ATM Interface TC Sublayer Table contains TC sublayer configuration and state parameters of those ATM interfaces which use TC sublayer for carrying ATM cells over SONET/SDH or DS3. This table has one entry per ATM interface port. This table is a sequence of atmInterfaceTCEntry which is a list that contains TC Sublayer parameters and state variables at the ATM interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- atmInterfaceTCTable - { atmMIBObjects 4 } 1.3.6.1.2.1.10.37.1.4
- atmInterfaceTCEntry - { atmInterfaceTCTable 1} 1.3.6.1.2.1.10.37.1.4.1

Table 129. atmInterfaceTCTable

Object	OID	Syntax	Access	Status	Supported
atmInterfaceOCDEvents	{ atmInterfaceTCEntry 1 }	Counter32	read-only	current	Y
atmInterfaceTCAlarmState	{ atmInterfaceTCEntry 2 }	INTEGER noAlarm(1) lcdFailure(2)	read-only	current	Y

5.13.4.3.1 atmInterfaceOCDEvents

The number of times the Out of Cell Delineation (OCD) events occur. If seven consecutive ATM cells have Header Error Control (HEC) violations, an OCD event occurs. A high number of OCD events may indicate a problem with the TC Sublayer.

5.13.4.3.2 atmInterfaceTCAlarmState

This variable indicates if there is an alarm present for the TC Sublayer. The value lcdFailure(2) indicates that the TC Sublayer is currently in the Loss of Cell Delineation (LCD) defect maintenance state. The value noAlarm(1) indicates that the TC Sublayer is currently not in the LCD defect maintenance state.

- noAlarm(1)
- lcdFailure(2)

5.13.4.4 atmTrafficDescrParamTable

ATM Traffic Descriptor Parameter Table contains a set of self-consistent ATM traffic parameters including the ATM traffic service category.

The ATM virtual link tables (that is, VPL and VCL tables) will use this ATM Traffic Descriptor table to assign traffic parameters and service category to the receive and transmit directions of the ATM virtual links (that is, VPLs and VCLs). The ATM VPL or VCL table will indicate a row in the atmTrafficDescrParamTable using its atmTrafficDescrParamIndex value. The management application can then compare a set of ATM traffic parameters with a single value.

If no suitable row(s) in the atmTrafficDescrParamTable exists, the manager must create a new row(s) in this table. If such a row is created, agent checks the sanity of that set of ATM traffic parameter values.

The manager may use atmTrafficDescrParamIndexNext in order to obtain a free atmTrafficDescrParamIndex value.

When creating a new row, the parameter values are checked for self-consistency. Predefined/template rows may be supported.

A row in the atmTrafficDescrParamTable is deleted by setting the atmTrafficDescrRowStatus to destroy(6). The agent will check whether this row is still in use by any entry of the atmVplTable or atmVclTable. The agent denies the request if the row is still in use.

This table is a sequence of atmTrafficDescrParamEntry which contains ATM traffic descriptor type and the associated parameters.

Index:

- atmTrafficDescrParamIndex

OIDs:

- atmTrafficDescrParamTable - { atmMIBObjects 5 } 1.3.6.1.2.1.10.37.1.5
- atmTrafficDescrParamEntry - { atmTrafficDescrParamTable 1} 1.3.6.1.2.1.10.37.1.5.1

Table 130. atmTrafficDescrParamTable

Object	OID	Syntax	Access	Status	Supported
atmTrafficDescrParamIndex	{ atmTrafficDescrParamEntry 1 }	AtmTrafficDescrParamIndex (1…2147483647)	not-accessible	current	N
atmTrafficDescrType	{ atmTrafficDescrParamEntry 2 }	OBJECT IDENTIFIER	read-create	current	N
atmTrafficDescrParam1	{ atmTrafficDescrParamEntry 3 }	Integer32	read-create	current	N
atmTrafficDescrParam2	{ atmTrafficDescrParamEntry 4 }	Integer32	read-create	current	N
atmTrafficDescrParam3	{ atmTrafficDescrParamEntry 5 }	Integer32	read-create	current	N
atmTrafficDescrParam4	{ atmTrafficDescrParamEntry 6 }	Integer32	read-create	current	N
atmTrafficDescrParam5	{ atmTrafficDescrParamEntry 7 }	Integer32	read-create	current	N
atmTrafficQoSClass	{ atmTrafficDescrParamEntry 8 }	INTEGER(0…255)	read-create	deprecated	N
atmTrafficDescrRowStatus	{ atmTrafficDescrParamEntry 9 }	RowStatus	read-create	current	N
atmServiceCategory	{ atmTrafficDescrParamEntry 10 }	AtmServiceCategory	read-create	current	N
atmTrafficFrameDiscard	{ atmTrafficDescrParamEntry 11 }	TruthValue	read-create	current	N

5.13.4.4.1 atmTrafficDescrParamIndex

This object is used by the virtual link table (that is, VPL or VCL table) to identify the row of this table. When creating a new row in the table the value of this index may be obtained by retrieving the value of atmTrafficDescrParamIndexNext.

5.13.4.4.2 atmTrafficDescrType

The value of this object identifies the type of ATM traffic descriptor. The type may indicate no traffic descriptor or traffic descriptor with one or more parameters. These parameters are specified as a parameter vector, in the corresponding instances of the objects:

- atmTrafficDescrParam1
- atmTrafficDescrParam2
- atmTrafficDescrParam3
- atmTrafficDescrParam4
- atmTrafficDescrParam5

The MIB recommended default value for this object is atmNoClpNoScr.

5.13.4.4.3 atmTrafficDescrParam1

The first parameter of the ATM traffic descriptor used according to the value of atmTrafficDescrType.

The MIB recommended default value for this object is { 0 }.

5.13.4.4.4 atmTrafficDescrParam2

The second parameter of the ATM traffic descriptor used according to the value of atmTrafficDescrType.

The MIB recommended default value for this object is { 0 }.

5.13.4.4.5 atmTrafficDescrParam3

The third parameter of the ATM traffic descriptor used according to the value of atmTrafficDescrType.

The MIB recommended default value for this object is { 0 }.

5.13.4.4.6 atmTrafficDescrParam4

The fourth parameter of the ATM traffic descriptor used according to the value of atmTrafficDescrType.

The MIB recommended default value for this object is { 0 }.

5.13.4.4.7 atmTrafficDescrParam5

The fifth parameter of the ATM traffic descriptor used according to the value of atmTrafficDescrType.

5.13.4.4.8 atmTrafficQoSClass

The value of this object identifies the QoS Class. Four Service classes have been specified in the ATM Forum UNI Specification:

1. Service Class A: Constant bit rate video and Circuit emulation

2. Service Class B: Variable bit rate video/audio

3. Service Class C: Connection-oriented data

4. Service Class D: Connectionless data

Four QoS classes numbered 1, 2, 3, and 4 have been specified with the aim to support service classes A, B, C, and D respectively.

An unspecified QoS Class numbered “0” is used for best effort traffic.

The MIB recommended default value for this object is { 0 }.

5.13.4.4.9 atmTrafficDescrRowStatus

This object is used to create a new row or modify or delete an existing row in this table.

The MIB recommended default value for this object is { active }.

5.13.4.4.10 atmServiceCategory

The ATM service category.

The MIB recommended default value for this object is { ubr }.

5.13.4.4.11 atmTrafficFrameDiscard

If set to “true,” this object indicates that the network is requested to treat data for this connection, in the given direction, as frames (for example, AAL5 CPCS_PDUs) rather than as individual cells. While the precise implementation is network-specific, this treatment may for example involve discarding entire frames during congestion, rather than a few cells from many frames.

The MIB recommended default value for this object is { true }.

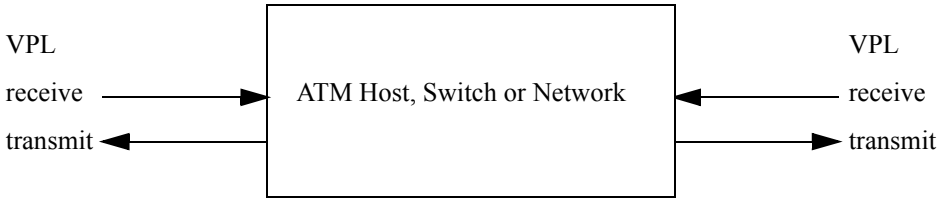
5.13.4.5 atmVplTable

ATM Interface Virtual Path Link (VPL) Table contains configuration and state information of a bi-directional Virtual Path Link (VPL). A bi-directional VPL is modeled as one entry in this table. This table can be used for PVCs, SVCs and Soft PVCs. Entries are not present in this table for the VPIs used by entries in the atmVclTable.

This table can be used to create, delete or modify a VPL that is terminated in an ATM host or switch. This table can also be used to create, delete or modify a VPL which is cross-connected to another VPL.

In the example below, the traffic flows on the receive and transmit directions of the VPLs are characterized by atmVplReceiveTrafficDescrIndex and atmVplTransmitTrafficDescrIndex respectively. The cross-connected VPLs are identified by atmVplCrossConnectIdentifier.

In the example below, the traffic flows on the receive and transmit directions of the VPLs are characterized by atmVplReceiveTrafficDescrIndex and,atmVplTransmitTrafficDescrIndex respectively.mThe cross-connected VPLs are identified by atmVplCrossConnectIdentifier.



This table is a sequence of atmVplEntry. The index for this table is the ifIndex and atmVplVpi objects.

To create a VPL at an ATM interface, either of the following procedures are used:

- Negotiated VPL establishment

1. The management application creates a VPL entry in the atmVplTable by setting atmVplRowStatus to createAndWait(5). This may fail for the following reasons:
   - The selected VPI value is unavailable
   - The selected VPI value is in use.

   Otherwise, the agent creates a row and reserves the VPI value on that port.
2. The manager selects an existing row(s) in the atmTrafficDescrParamTable, thereby, selecting a set of self-consistent ATM traffic parameters and the service category for receive and transmit directions of the VPL.

2a.If no suitable row(s) in the atmTrafficDescrParamTable exists, the manager must create a new row(s) in that table.

2b.The manager characterizes the VPL's traffic parameters through setting the atmVplReceiveTrafficDescrIndex and the atmVplTransmitTrafficDescrIndex values in the VPL table, which point to the rows containing desired ATM traffic parameter values in the atmTrafficDescrParamTable. The agent will check the availability of resources and may refuse the request. If the transmit and receive service categories are inconsistent, the agent should refuse the request.

3. The manager activates the VPL by setting the atmVplRowStatus to active(1). If this set is successful, the agent has reserved the resources to satisfy the requested traffic parameter values and the service category for that VPL.
4. If the VPL terminates a VPC in the ATM host or switch, the manager turns on the atmVplAdminStatus to up(1) to turn the VPL traffic flow on. Otherwise, the atmVpCrossConnectTable must be used to cross-connect the VPL to another VPL(s) in an ATM switch or network.

- One-Shot VPL Establishment

  A VPL may also be established in one step by a set-request with all necessary VPL parameter values and atmVplRowStatus set to createAndGo(4). In contrast to the negotiated VPL establishment which allows for detailed error checking (that is, set errors are explicitly linked to particular resource acquisition failures), the one-shot VPL establishment performs the setup on one operation but does not have the advantage of step-wise error checking.

- VPL Retirement

  A VPL is released by setting atmVplRowStatus to destroy(6), and the agent may release all associated resources.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- atmVplVpi

OIDs:

- atmVplTable - { atmMIBObjects 6 } 1.3.6.1.2.1.10.37.1.6
- atmVplEntry - { atmVplTable 1} 1.3.6.1.2.1.10.37.1.6.1

Table 131. atmVplTable

Object	OID	Syntax	Access	Status	Supported
atmVplVpi	{ atmVplEntry 1 }	AtmVpIdentifier	not-accessible	current	N
atmVplAdminStatus	{ atmVplEntry 2 }	AtmVorXAdminStatus	read-create	current	N
atmVplOperStatus	{ atmVplEntry 3 }	AtmVorXOperStatus	read-only	current	N
atmVplLastChange	{ atmVplEntry 4 }	AtmVorXLastChange	read-only	current	N
atmVplReceiveTrafficDescrIndex	{ atmVplEntry 5 }	AtmTrafficDescrParamIndex	read-create	current	N
atmVplTransmitTrafficDescrIndex	{ atmVplEntry 6 }	AtmTrafficDescrParamIndex	read-create	current	N
atmVplCrossConnectIdentifier	{ atmVplEntry 7 }	INTEGER(1…2147483647)	read-create	current	N
atmVplRowStatus	{ atmVplEntry 8 }	RowStatus	read-create	current	N
atmVplCastType	{ atmVplEntry 9 }	AtmConnCastType	read-create	current	N
atmVplConnKind	{ atmVplEntry 10 }	AtmConnKind	read-create	current	N

5.13.4.5.1 atmVplVpi

The VPI value of the VPL.

5.13.4.5.2 atmVplAdminStatus

This object is instantiated only for a VPL which terminates a VPC (that is, one which is NOT cross-connected to other VPLs). Its value specifies the desired administrative state of the VPL.

The MIB recommended default value for this object is { down }.

5.13.4.5.3 atmVplOperStatus

The current operational status of the VPL.

5.13.4.5.4 atmVplLastChange

The value of sysUpTime at the time this VPL entered its current operational state.

5.13.4.5.5 atmVplReceiveTrafficDescrIndex

The value of this object identifies the row in the atmTrafficDescrParamTable which applies to the receive direction of the VPL.

The MIB recommended default value for this object is { 0 }.

5.13.4.5.6 atmVplTransmitTrafficDescrIndex

The value of this object identifies the row in the atmTrafficDescrParamTable which applies to the transmit direction of the VPL.

The MIB recommended default value for this object is { 0 }.

5.13.4.5.7 atmVplCrossConnectIdentifier

This object is instantiated only for a VPL which is cross-connected to other VPLs that belong to the same VPC. All such associated VPLs have the same value of this object, and all their cross-connections are identified either by entries that are indexed by the same value of atmVpCrossConnectIndex in the atmVpCrossConnectTable of this MIB module or by the same value of the cross-connect index in the cross-connect table for SVCs and Soft PVCs (defined in a separate MIB module). At no time should entries in these respective cross-connect tables exist simultaneously with the same cross-connect index value. The value of this object is initialized by the agent after the associated entries in the atmVpCrossConnectTable have been created.

5.13.4.5.8 atmVplRowStatus

This object is used to create, delete or modify a row in this table. To create a new VCL, this object is initially set to “createAndWait” or “createAndGo”. This object should not be set to “active” unless the following columnar objects have been set to their desired value in this row: atmVplReceiveTrafficDescrIndex and atmVplTransmitTrafficDescrIndex. The DESCRIPTION of atmVplEntry provides further guidance to row treatment in this table.

The MIB recommended default value for this object is { createAndWait }.

5.13.4.5.9 atmVplCastType

The connection topology type.

The MIB recommended default value for this object is { p2p }.

5.13.4.5.10 atmVplConnKind

The use of call control.

The MIB recommended default value for this object is { pvc }.

5.13.4.6 atmVclTable

The ATM Interface Virtual Channel Link (VCL) Table contains configuration and state information of a bi-directional Virtual Channel Link (VCL) at an ATM interface. A bi-directional VCL is modeled as one entry in this table. This table can be used for PVCs, SVCs and Soft PVCs. This table can be used to create, delete or modify a VCL that is terminated in an ATM host or switch. This table can also be used to create, delete or modify a VCL that is cross-connected to another VCL.

This table is a sequence of atmVclEntry and indexed by ifIndex, atmVclVpi, and atmVclVci objects.

To create a VCL at an ATM interface, either of the following procedures are used:

- Negotiated VCL establishment

1. The management application creates a VCL entry in the atmVclTable by setting atmVclRowStatus to createAndWait(5). This may fail for the following reasons:
    - •The selected VPI/VCI values are unavailable
    - •The selected VPI/VCI values are in use

Otherwise, the agent creates a row and reserves the VPI/VCI values on that port.

2. The manager selects an existing row(s) in the atmTrafficDescrParamTable, thereby, selecting a set of self-consistent ATM traffic parameters and the service category for receive and transmit directions of the VCL.

2a.If no suitable row(s) in the atmTrafficDescrParamTable exists, the manager must create a new row(s) in that table.

2b.The manager characterizes the VCL's traffic parameters through setting the atmVclReceiveTrafficDescrIndex and the atmVclTransmitTrafficDescrIndex values in the VCL table, which point to the rows containing desired ATM traffic parameter values in the atmTrafficDescrParamTable. The agent will check the availability of resources and may refuse the request. If the transmit and receive service categories are inconsistent, the agent should refuse the request.

3. The manager activates the VCL by setting the atmVclRowStatus to active(1) (for requirements on this activation see the description of atmVclRowStatus). If this set is successful, the agent has reserved the resources to satisfy the requested traffic parameter values and the service category for that VCL.
4. If the VCL terminates a VCC in the ATM host or switch, the manager turns on the atmVclAdminStatus to up(1) to turn the VCL traffic flow on. Otherwise, the atmVcCrossConnectTable must be used to cross-connect the VCL to another VCL(s) in an ATM switch or network.
   - One-Shot VCL Establishment

     A VCL may also be established in one step by a set-request with all necessary VCL parameter values and atmVclRowStatus set to createAndGo(4).

     In contrast to the negotiated VCL establishment which allows for detailed error checking (that is, set errors are explicitly linked to particular resource acquisition failures), the one-shot VCL establishment performs the setup on one operation but does not have the advantage of step-wise error checking.
   - VCL Retirement

     A VCL is released by setting atmVclRowStatus to destroy(6), and the agent may release all associated resources.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- atmVclVpi
- atmVclVci

OIDs:

- atmVclTable - { atmMIBObjects 7 } 1.3.6.1.2.1.10.37.1.7
- atmVclEntry - { atmVclTable 1} 1.3.6.1.2.1.10.37.1.7.1

Table 132. atmVclTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
atmVclVpi	{ atmVclEntry 1 }	AtmVpIdentifier	not-accessible	current	Y
atmVclVci	{ atmVclEntry 2 }	AtmVcIdentifier	not-accessible	current	Y
atmVclAdminStatus	{ atmVclEntry 3 }	AtmVorXAdminStatus	read-create	current	Y
atmVclOperStatus	{ atmVclEntry 4 }	AtmVorXOperStatus	read-only	current	Y
atmVclLastChange	{ atmVclEntry 5 }	AtmVorXLastChange	read-only	current	Y
atmVclReceiveTrafficDescrIndex	{ atmVclEntry 6 }	AtmTrafficDescrParamIndex	read-create	current	N
atmVclTransmitTrafficDescrIndex	{ atmVclEntry 7 }	AtmTrafficDescrParamIndex	read-create	current	N

Table 132. atmVclTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
atmVccAalType	{ atmVclEntry 8 }	INTEGER aal1(1) aal34(2) aal5(3) other(4) unknown(5) aal2(6)	read-create	current	Y
atmVccAal5CpcsTransmitSduSize	{ atmVclEntry 9 }	INTEGER(1…65535)	read-create	current	Y
atmVccAal5CpcsReceiveSduSize	{ atmVclEntry 10 }	INTEGER(1…65535)	read-create	current	Y
atmVccAal5EncapsType	{ atmVclEntry 11 }	INTEGER vcMultiplexRoutedProtocol(1) vcMultiplexBridgedProtocol8023(2) vcMultiplexBridgedProtocol8025(3) vcMultiplexBridgedProtocol(4) vcMultiplexLANemulation8023(5) vcMultiplexLANemulation8025(6) llcEncapsulation(7) multiprotocolFrameRelaySscs(8) other(9) unknown(10)	read-create	current	Y
atmVclCrossConnectIdentifier	{ atmVclEntry 12 }	INTEGER(1…2147483674)	read-only	current	N
atmVclRowStatus	{ atmVclEntry 13 }	RowStatus	read-create	current	Y
atmVclCastType	{ atmVclEntry 14 }	AtmConnCastType	read-create	current	N
atmVclConnKind	{ atmVclEntry 15 }	AtmConnKind	read-create	current	N

5.13.4.6.1 atmVclVpi

The VPI value of the VPL.

5.13.4.6.2 atmVclVci

The VCI value of the VCL.

5.13.4.6.3 atmVclAdminStatus

This object is instantiated only for a VCL which terminates a VCC (that is, one which is NOT cross-connected to other VCLs). Its value specifies the desired administrative state of the VCL.

The MIB recommended default value for this object is { down }.

5.13.4.6.4 atmVclOperStatus

The current operational status of the VCL.

5.13.4.6.5 atmVclLastChange

The value of sysUpTime at the time this VCL entered its current operational state.

5.13.4.6.6 atmVclReceiveTrafficDescrIndex

The value of this object identifies the row in the ATM Traffic Descriptor Table which applies to the receive direction of this VCL.

The MIB recommended default value for this object is { 0 }.

5.13.4.6.7 atmVclTransmitTrafficDescrIndex

The value of this object identifies the row of the ATM Traffic Descriptor Table which applies to the transmit direction of this VCL.

The MIB recommended default value for this object is { 0 }.

5.13.4.6.8 atmVccAalType

An instance of this object only exists when the local VCL end-point is also the VCC end-point, and AAL is in use. The type of AAL used on this VCC. The AAL type includes AAL1, AAL2, AAL3/4, and AAL5. The other(4) may be user-defined AAL type. The unknown type indicates that the AAL type cannot be determined.

- aal1(1)
- aal34(2)
- aal5(3)
- other(4)
- unknown(5)
- aal2(6)

The MIB recommended default value for this object is { aal5 }.

Valid Object Value(s)

This object will always return aal5(3) and attempts to write a different value will return notWritable(17) in the error-status filed of the SNMP PDU.

5.13.4.6.9 atmVccAal5CpcsTransmitSduSize

An instance of this object only exists when the local VCL end-point is also the VCC end-point, and AAL5 is in use. The maximum AAL5 CPCS SDU size in octets that is supported on the transmit direction of this VCC.

The MIB recommended default value for this object is { 9188 }.

Valid Object Value(s)

This object will always return 1536 and attempts to write a different value will return notWritable(17) in the error-status filed of the SNMP PDU.

5.13.4.6.10 atmVccAal5CpcsReceiveSduSize

An instance of this object only exists when the local VCL end-point is also the VCC end-point, and AAL5 is in use. The maximum AAL5 CPCS SDU size in octets that is supported on the receive direction of this VCC.

The MIB recommended default value for this object is { 9188 }.

Valid Object Value(s)

This object will always return 1536 and attempts to write a different value will return notWritable(17) in the error-status filed of the SNMP PDU.

5.13.4.6.11 atmVccAal5EncapsType

An instance of this object only exists when the local VCL end-point is also the VCC end-point, and AAL5 is in use. The type of data encapsulation used over the AAL5 SSCS layer. The definitions reference RFC 1483 Multiprotocol Encapsulation over ATM AAL5 and to the ATM Forum LAN Emulation specification.

- vcMultiplexRoutedProtocol(1)
- vcMultiplexBridgedProtocol8023(2)

- vcMultiplexBridgedProtocol8025(3)
- vcMultiplexBridgedProtocol8026(4)
- vcMultiplexLANemulation8023(5)
- vcMultiplexLANemulation8025(6)
- llcEncapsulation(7)
- multiprotocolFrameRelaySscs(8)
- other(9)
- unknown(10)

The MIB recommended default value for this object is { llcEncapsulation }.

Valid Object Values

- vcMultiplexBridgedProtocol8023(2)
- llcEncapsulation(7)

5.13.4.6.12 atmVclCrossConnectIdentifier

This object is instantiated only for a VCL which is cross-connected to other VCLs that belong to the same VCC. All such associated VCLs have the same value of this object, and all their cross-connections are identified either by entries that are indexed by the same value of atmVcCrossConnectIndex in the atmVcCrossConnectTable of this MIB module or by the same value of the cross-connect index in the cross-connect table for SVCs and Soft PVCs (defined in a separate MIB module).

At no time should entries in these respective cross-connect tables exist simultaneously with the same cross-connect index value. The value of this object is initialized by the agent after the associated entries in the atmVcCrossConnectTable have been created.

5.13.4.6.13 atmVclRowStatus

This object is used to create, delete or modify a row in this table. To create a new VCL, this object is initially set to “createAndWait” or “createAndGo.” This object should not be set to “active” unless the following columnar objects have been set to their desired value in this row: atmVclReceiveTrafficDescrIndex, atmVclTransmitTrafficDescrIndex. In addition, if the local VCL end-point is also the VCC end-point: atmVccAalType. In addition, for AAL5 connections only: atmVccAal5CpcsTransmitSduSize, atmVccAal5CpcsReceiveSduSize, and atmVccAal5EncapsType. (The existence of these objects imply the AAL connection type.). The DESCRIPTION of atmVclEntry provides further guidance to row treatment in this table.

Valid Object Value(s)

The MIB recommended default value for this object is { createAndGo }.

The { createAndWait } RowStatus (Section 4.6, Page 13) is not supported.

5.13.4.6.14 atmVclCastType

The connection topology type.

The MIB recommended default value for this object is { p2p }.

5.13.4.6.15 atmVclConnKind

The use of call control.

The MIB recommended default value for this object is { pvc }.

5.13.4.7 atmVpCrossConnectIndexNext

This object contains an appropriate value to be used for atmVpCrossConnectIndex when creating entries in the atmVpCrossConnectTable. The value 0 indicates that no unassigned entries are available. To obtain the atmVpCrossConnectIndex value for a new entry, the manager issues a management protocol retrieval operation to obtain the current value of this object. After each retrieval, the agent should modify the value to the next unassigned

index. After a manager retrieves a value the agent will determine through its local policy when this index value is made available for reuse.

OID:

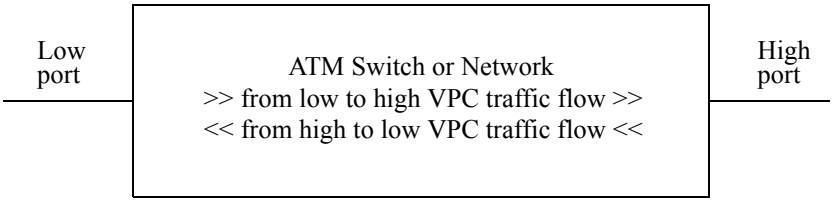
- atmVpCrossConnectIndexNext - { atmMIBObjects 8 }

5.13.4.8 atmVpCrossConnectTable

The ATM Virtual Path (VP) Cross Connect Table contains configuration and state information of point-to-point, point-to-multipoint, or multipoint-to-multipoint VP cross-connects for PVCs. An entry in this table models two cross-connected VPLs. Each VPL must have its atmConnKind set to pvc(1).

This table has read-create access and can be used to cross-connect the VPLs together in an ATM switch or network. The atmVpCrossConnectIndex is used to associate the related VPLs that are cross-connected together.

The ATM VP Cross Connect Table models each bi-directional VPC cross-connect as a set of entries in the atmVpCrossConnectTable. A point-to-point VPC cross-connect is modeled as one entry; a point-to-multipoint (N leafs) VPC cross-connect as N entries in this table; and a multipoint-to-multipoint (N parties) VPC cross- connect as N(N-1)/2 entries in this table. In the latter cases, all the N (or N(N-1)/2) entries are associated with a single VPC cross-connect by having the same value of atmVpCrossConnectIndex.



The terms low and high are chosen to represent numerical ordering of the two interfaces associated with a VPC cross-connect. That is, the ATM interface with the lower value of ifIndex is termed “low” while the other ATM interface associated with the, VPC cross-connect is termed “high.” This terminology is used to provide directional information; for example, the atmVpCrossConnectL2HOperStatus applies to the low->high direction, and atmVpCrossConnectH2LOperStatus applies to the high->low direction, as illustrated above.

The atmVpCrossConnectTable is a sequence of atmVpCrossConnectEntry and is index by atmVpCrossConnectIndex, atmVpCrossConnectLowIfIndex, atmVpCrossConnectLowVpi, atmVpCrossConnectHighIfIndex, and atmVpCrossConnectHighVpi objects.

- Step-wise Procedures to set up a VP Cross-connect

  Once the entries in the atmVplTable are created, the following procedures are used to cross-connect the VPLs together.

1. The manager obtains a unique atmVpCrossConnectIndex by reading the atmVpCrossConnectIndexNext object.
2. Next, the manager creates a set of one or more rows in the ATM VP Cross Connect Table, one for each cross-connection between two VPLs. Each row is indexed by the ATM interface port numbers and VPI values of the two ends of that cross-connection. This set of rows specifies the topology of the VPC cross-connect and is identified by a single value of atmVpCrossConnectIndex.
   - Negotiated VP Cross-Connect Establishment

2a.The manager creates a row in this table by setting atmVpCrossConnectRowStatus to createAndWait(5). The agent checks the requested topology and the mutual sanity of the ATM traffic parameters and service categories, that is, the row creation fails if:

- •the requested topology is incompatible with associated values of atmVplCastType,
- •the requested topology is not supported by the agent,
- •the traffic/service category parameter values associated with the requested row are incompatible with those of already existing rows for this VP cross-connect. [For example, for setting up a point-to-point VP cross-connect, the ATM traffic parameters in the receive direction of a VPL at the low end of the cross-connect must equal to the traffic parameters in the transmit direction of the other VPL at the high end of the cross-connect, otherwise, the row creation fails.] The agent also checks for internal errors in building the cross-connect.

The atmVpCrossConnectIndex values in the corresponding atmVplTable rows are filled in by the agent at this point.

2b.The manager promotes the row in the atmVpCrossConnectTable by setting atmVpCrossConnectRowStatus to active(1). If this set is successful, the agent has reserved the resources specified by the ATM traffic parameter and Service category values for each direction of the VP cross-connect in an ATM switch or network.

3. The manager sets the atmVpCrossConnectAdminStatus to up(1) in all rows of this VP cross-connect to turn the traffic flow on.
   - One-Shot VP Cross-Connect Establishment

     A VP cross-connect may also be established in one step by a set-request with all necessary parameter values and atmVpCrossConnectRowStatus set to createAndGo(4).

     In contrast to the negotiated VP cross-connect establishment which allows for detailed error checking (that is, set errors are explicitly linked to particular resource acquisition failures), the one-shot VP cross-connect establishment performs the setup on one operation but does not have the advantage of step-wise error checking.

   - VP Cross-Connect Retirement

     A VP cross-connect identified by a particular value of atmVpCrossConnectIndex is released by:

4. Setting atmVpCrossConnectRowStatus of all rows identified by this value of atmVpCrossConnectIndex to destroy(6). The agent may release all associated resources, and the atmVpCrossConnectIndex values in the corresponding atmVplTable row are removed. Note that a situation when only a subset of the associated rows are deleted corresponds to a VP topology change.
5. After deletion of the appropriate atmVpCrossConnectEntries, the manager may set atmVplRowStatus to destroy(6) the associated VPLs. The agent releases the resources and removes the associated rows in the atmVplTable.
   - VP Cross-connect Reconfiguration

     At the discretion of the agent, a VP cross-connect may be reconfigured by adding and/or deleting leafs to/ from the VP topology as per the VP cross-connect establishment/retirement procedures. Reconfiguration of traffic/service category parameter values requires release of the VP cross-connect before those parameter values may by changed for individual VPLs.

Indices:

- atmVpCrossConnectIndex
- atmVpCrossConnectLowIfIndex
- atmVpCrossConnectLowVpi
- atmVpCrossConnectHighIfIndex
- atmVpCrossConnectHighVpi

OIDs:

- atmVpCrossConnectTable - { atmMIBObjects 9 } 1.3.6.1.2.1.10.37.1.9
- atmVpCrossConnectEntry - { atmVpCrossConnectTable 1} 1.3.6.1.2.1.10.37.1.9.1

Table 133. atmVpCrossConnectTable

Object	OID	Syntax	Access	Status	Supported
atmVpCrossConnectIndex	{ atmVpCrossConnectEntry 1 }	INTEGER(1…2147483647)	not-accessible	current	N
atmVpCrossConnectLowIfIndex	{ atmVpCrossConnectEntry 2 }	InterfaceIndex	not-accessible	current	N
atmVpCrossConnectLowVpi	{ atmVpCrossConnectEntry 3 }	AtmVpIdentifier	not-accessible	current	N
atmVpCrossConnectHighIfIndex	{ atmVpCrossConnectEntry 4 }	InterfaceIndex	not-accessible	current	N
atmVpCrossConnectHighVpi	{ atmVpCrossConnectEntry 5 }	AtmVpIdentifier	not-accessible	current	N
atmVpCrossConnectAdminStatus	{ atmVpCrossConnectEntry 6 }	AtmVorXAdminStatus	read-create	current	N
atmVpCrossConnectL2HOperStatus	{ atmVpCrossConnectEntry 7 }	AtmVorXOperStatus	read-only	current	N
atmVpCrossConnectH2LOperStatus	{ atmVpCrossConnectEntry 8 }	AtmVorXOperStatus	read-only	current	N
atmVpCrossConnectL2HLastChange	{ atmVpCrossConnectEntry 9 }	AtmVorXLastChange	read-only	current	N
atmVpCrossConnectH2LLastChange	{ atmVpCrossConnectEntry 10 }	AtmVorXLastChange	read-only	current	N
atmVpCrossConnectRowStatus	{ atmVpCrossConnectEntry 11 }	RowStatus	read-create	current	N

5.13.4.8.1 atmVpCrossConnectIndex

A unique value to identify this VP cross-connect. For each VPL associated with this cross-connect, the agent reports this cross-connect index value in the atmVplCrossConnectIdentifier attribute of the corresponding atmVplTable entries.

5.13.4.8.2 atmVpCrossConnectLowIfIndex

The ifIndex value of the ATM interface for this VP cross-connect. The term low implies that this ATM interface has the numerically lower ifIndex value than the other ATM interface identified in the same atmVpCrossConnectEntry.

5.13.4.8.3 atmVpCrossConnectLowVpi

The VPI value at the ATM interface associated with the VP cross-connect that is identified by atmVpCrossConnectLowIfIndex.

5.13.4.8.4 atmVpCrossConnectHighIfIndex

The ifIndex value of the ATM interface for this VP cross-connect. The term high implies that this ATM interface has the numerically higher ifIndex value than the other ATM interface identified in the same atmVpCrossConnectEntry.

5.13.4.8.5 atmVpCrossConnectHighVpi

The VPI value at the ATM interface associated with the VP cross-connect that is identified by atmVpCrossConnectHighIfIndex.

5.13.4.8.6 atmVpCrossConnectAdminStatus

The desired administrative status of this bi-directional VP cross-connect.

The MIB recommended default value for this object is { down }.

5.13.4.8.7 atmVpCrossConnectL2HOperStatus

The operational status of the VP cross-connect in one direction; (that is, from the low to high direction).

5.13.4.8.8 atmVpCrossConnectH2LOperStatus

The operational status of the VP cross-connect in one direction; (that is, from the high to low direction).

5.13.4.8.9 atmVpCrossConnectL2HLastChange

The value of sysUpTime at the time this VP cross-connect entered its current operational state in the low to high direction.

5.13.4.8.10 atmVpCrossConnectH2LLastChange

The value of sysUpTime at the time this VP cross-connect entered its current operational in the high to low direction.

5.13.4.8.11 atmVpCrossConnectRowStatus

The status of this entry in the atmVpCrossConnectTable. This object is used to create a cross-connect for cross-connecting VPLs which are created using the atmVplTable or to change or delete an existing cross-connect. This object must be initially set to “createAndWait” or “createAndGo.” To turn on a VP cross-connect, the atmVpCrossConnectAdminStatus is set to “up.”

The MIB recommended default value for this object is { createAndWait }.

5.13.4.9 atmVcCrossConnectIndexNext

This object contains an appropriate value to be used for atmVcCrossConnectIndex when creating entries in the atmVcCrossConnectTable. The value 0 indicates that no unassigned entries are available. To obtain the atmVcCrossConnectIndex value for a new entry, the manager issues a management protocol retrieval operation to obtain the current value of this object. After each retrieval, the agent should modify the value to the next unassigned index. After a manager retrieves a value the agent will determine through its local policy when this index value is made available for reuse.

OID:

- atmVcCrossConnectIndexNext - { atmMIBObjects 10 } 1.3.6.1.2.1.10.37.1.10

5.13.4.10 atmVcCrossConnectTable

The ATM Virtual Channel (VC) Cross Connect Table contains configuration and state information of point-to-point, point-to-multipoint or multipoint-to-multipoint VC cross-connects for PVCs. An entry in the ATM VC Cross Connect table is used to model a bi-directional ATM VC cross-connect cross-connecting two end points

This table has read-create access and is used to cross-connect the VCLs together in an ATM switch or network that belong to a VC connection. The atmVcCrossConnectIndex is used to associate the related VCLs that are cross-connected together.

The model using step-wise procedures described for setting up a VP cross-connect is also used for setting up a VC cross-connect.

This table is a sequence of atmVcCrossConnectEntry and is indexed by atmVcCrossConnectIndex, atmVcCrossConnectLowIfIndex, atmVcCrossConnectLowVpi, atmVcCrossConnectLowVci, atmVcCrossConnectHighIfIndex, atmVcCrossConnectHighVpi, and atmVcCrossConnectHighVci objects.

An entry in this table models two cross-connected VCLs. Each VCL must have its atmConnKind set to pvc(1).

- Step-wise Procedures to set up a VC Cross-connect

  Once the entries in the atmVclTable are created, the following procedures are used to cross-connect the VCLs together to form a VCC segment.

1. The manager obtains a unique atmVcCrossConnectIndex by reading the atmVcCrossConnectIndexNext object.
2. Next, the manager creates a set of one or more rows in the ATM VC Cross Connect Table, one for each cross-connection between two VCLs. Each row is indexed by the ATM interface port numbers and VPI/VCI values of the two ends of that cross-connection. This set of rows specifies the topology of the VCC cross-connect and is identified by a single value of atmVcCrossConnectIndex.
   - Negotiated VC Cross-Connect Establishment

     2a. The manager creates a row in this table by setting atmVcCrossConnectRowStatus to createAndWait(5). The agent checks the requested topology and the mutual sanity of the ATM traffic parameters and service categories, that is, the row creation fails if:

•the requested topology is incompatible with associated values of atmVclCastType,

•the requested topology is not supported by the agent,

•the traffic/service category parameter values associated with the requested row are incompatible with those of already existing rows for this VC cross-connect.

[For example, for setting up a point-to-point VC cross-connect, the ATM traffic parameters in the receive direction of a VCL at the low end of the cross-connect must equal to the traffic parameters in the transmit direction of the other VCL at the high end of the cross-connect, otherwise, the row creation fails.] The agent also checks for internal errors in building the cross-connect.

The atmVcCrossConnectIndex values in the corresponding atmVclTable rows are filled in by the agent at this point.

2b. The manager promotes the row in the atmVcCrossConnectTable by setting atmVcCrossConnectRowStatus to active(1). If this set is successful, the agent has reserved the resources specified by the ATM traffic parameter and Service category values for each direction of the VC cross-connect in an ATM switch or network.

3. The manager sets the atmVcCrossConnectAdminStatus to up(1) in all rows of this VC cross-connect to turn the traffic flow on.

- One-Shot VC Cross-Connect Establishment

  A VC cross-connect may also be established in one step by a set-request with all necessary parameter values and atmVcCrossConnectRowStatus set to createAndGo(4).

  In contrast to the negotiated VC cross-connect establishment which allows for detailed error checking that is, set errors are explicitly linked to particular resource acquisition failures), the one-shot VC cross-connect establishment performs the setup on one operation but does not have the advantage of step-wise error checking.

- VC Cross-Connect Retirement

  A VC cross-connect identified by a particular value of atmVcCrossConnectIndex is released by:

4. Setting atmVcCrossConnectRowStatus of all rows identified by this value of atmVcCrossConnectIndex to destroy(6). The agent may release all associated resources, and the atmVcCrossConnectIndex values in the corresponding atmVclTable row are removed. Note that a situation when only a subset of the associated rows are deleted corresponds to a VC topology change.

5. After deletion of the appropriate atmVcCrossConnectEntries, the manager may set atmVclRowStatus to destroy(6) the associated VCLs. The agent releases the resources and removes the associated rows in the atmVclTable.

- VC Cross-Connect Reconfiguration

  At the discretion of the agent, a VC cross-connect may be reconfigured by adding and/or deleting leafs to/ from the VC topology as per the VC cross-connect establishment/retirement procedures. Reconfiguration of traffic/service category parameter values requires release of the VC cross-connect before those parameter values may by changed for individual VCLs.

Indices:

- atmVcCrossConnectIndex
- atmVcCrossConnectLowIfIndex
- atmVcCrossConnectLowVpi
- atmVcCrossConnectLowVci
- atmVcCrossConnectHighIfIndex
- atmVcCrossConnectHighVpi
- atmVcCrossConnectHighVci

OIDs:

- atmVcCrossConnectTable - { atmMIBObjects 11 } 1.3.6.1.2.1.10.37.1.11
- atmVcCrossConnectEntry - { atmVcCrossConnectTable 1} 1.3.6.1.2.1.10.37.1.11.1

Table 134. atmVcCrossConnectTable

Object	OID	Syntax	Access	Status	Supported
atmVcCrossConnectIndex	{ atmVcCrossConnectEntry 1 }	INTEGER(1…2147483647)	not-accessible	current	N
atmVcCrossConnectLowIfIndex	{ atmVcCrossConnectEntry 2 }	InterfaceIndex	not-accessible	current	N
atmVcCrossConnectLowVpi	{ atmVcCrossConnectEntry 3 }	AtmVpIdentifier	not-accessible	current	N
atmVcCrossConnectLowVci	{ atmVcCrossConnectEntry 4 }	AtmVcIdentifier	not-accessible	current	N
atmVcCrossConnectHighIfIndex	{ atmVcCrossConnectEntry 5 }	InterfaceIndex	not-accessible	current	N
atmVcCrossConnectHighVpi	{ atmVcCrossConnectEntry 6 }	AtmVpIdentifier	not-accessible	current	N
atmVcCrossConnectHighVci	{ atmVcCrossConnectEntry 7 }	AtmVcIdentifier	not-accessible	current	N
atmVcCrossConnectAdminStatus	{ atmVcCrossConnectEntry 8 }	AtmVorXAdminStatus	read-create	current	N
atmVcCrossConnectL2HOperStatus	{ atmVcCrossConnectEntry 9 }	AtmVorXOperStatus	read-only	current	N
atmVcCrossConnectH2LOperStatus	{ atmVcCrossConnectEntry 10 }	AtmVorXOperStatus	read-only	current	N
atmVcCrossConnectL2HLastChange	atmVcCrossConnectEntry 11 }	AtmVorXLastChange	read-only	current	N
atmVcCrossConnectH2LLastChange	atmVcCrossConnectEntry 12 }	AtmVorXLastChange	read-only	current	N
atmVcCrossConnectRowStatus	atmVcCrossConnectEntry 13 }	RowStatus	read-create	current	N

5.13.4.10.1 atmVcCrossConnectIndex

A unique value to identify this VC cross-connect. For each VCL associated with this cross-connect, the agent reports this cross-connect index value in the atmVclCrossConnectIdentifier attribute of the corresponding atmVclTable entries.

5.13.4.10.2 atmVcCrossConnectLowIfIndex

The ifIndex value of the ATM interface for this VC cross-connect. The term low implies that this ATM interface has the numerically lower ifIndex value than the other ATM interface identified in the same atmVcCrossConnectEntry.

5.13.4.10.3 atmVcCrossConnectLowVpi

The VPI value at the ATM interface associated with the VC cross-connect that is identified by atmVcCrossConnectLowIfIndex.

5.13.4.10.4 atmVcCrossConnectLowVci

The VCI value at the ATM interface associated with this VC cross-connect that is identified by atmVcCrossConnectLowIfIndex.

5.13.4.10.5 atmVcCrossConnectHighIfIndex

The ifIndex value for the ATM interface for this VC cross-connect. The term high implies that this ATM interface has the numerically higher ifIndex value than the other ATM interface identified in the same atmVcCrossConnectEntry.

5.13.4.10.6 atmVcCrossConnectHighVpi

The VPI value at the ATM interface associated with the VC cross-connect that is identified by atmVcCrossConnectHighIfIndex.

5.13.4.10.7 atmVcCrossConnectHighVci

The VCI value at the ATM interface associated with the VC cross-connect that is identified by atmVcCrossConnectHighIfIndex.

5.13.4.10.8 atmVcCrossConnectAdminStatus

The desired administrative status of this bi-directional VC cross-connect.

The MIB recommended default value for this object is { down }.

5.13.4.10.9 atmVcCrossConnectL2HOperStatus

The current operational status of the VC cross-connect in one direction; (that is, from the low to high direction).

5.13.4.10.10 atmVcCrossConnectH2LOperStatus

The current operational status of the VC cross-connect in one direction; (that is, from the high to low direction).

5.13.4.10.11 atmVcCrossConnectL2HLastChange

The value of sysUpTime at the time this VC cross-connect entered its current operational state in low to high direction.

5.13.4.10.12 atmVcCrossConnectH2LLastChange

The value of sysUpTime at the time this VC cross-connect entered its current operational state in high to low direction.

5.13.4.10.13 atmVcCrossConnectRowStatus

The status of this entry in the atmVcCrossConnectTable. This object is used to create a new cross-connect for cross-connecting VCLs which are created using the atmVclTable or to change or delete existing cross-connect. This object must be initially set to “createAndWait” or “createAndGo.” To turn on a VC cross-connect, the atmVcCrossConnectAdminStatus is set to “up.”

The MIB recommended default value for this object is { createAndWait }.

5.13.4.11 aal5VccTable

The AAL5 Virtual Channel Connection Performance Statistics Table contains the AAL5 performance statistics of a VCC at the interface associated with an AAL5 entity in an ATM host or ATM switch.

This table is a sequence of aal5VccEntry which contains the AAL5 VCC performance parameters.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- aal5VccVpi
- aal5VccVci

OIDs:

- aal5VccTable - { atmMIBObjects 12 } 1.3.6.1.2.1.10.37.1.12
- aal5VccEntry - { aal5VccTable 1} 1.3.6.1.2.1.10.37.1.12.1

Table 135. aal5VccTable

Object	OID	Syntax	Access	Status	Supported
aal5VccVpi	{ aal5VccEntry 1 }	AtmVpIdentifier	not-accessible	current	Y
aal5VccVci	{ aal5VccEntry 2 }	AtmVcIdentifier	not-accessible	current	Y
aal5VccCrcErrors	{ aal5VccEntry 3 }	Counter32	read-only	current	Y

Table 135. aal5VccTable

Object	OID	Syntax	Access	Status	Supported
aal5VccSarTimeOuts	{ aal5VccEntry 4 }	Counter32	read-only	current	Y
aal5VccOverSizedSDUs	{ aal5VccEntry 5 }	Counter32	read-only	current	Y

5.13.4.11.1 aal5VccVpi

The VPI value of the AAL5 VCC at the interface identified by the ifIndex.

5.13.4.11.2 aal5VccVci

The VCI value of the AAL5 VCC at the interface identified by the ifIndex.

5.13.4.11.3 aal5VccCrcErrors

The number of AAL5 CPCS PDUs received with CRC-32 errors on this AAL5 VCC at the interface associated with an AAL5 entity.

5.13.4.11.4 aal5VccSarTimeOuts

The number of partially re-assembled AAL5 CPCS PDUs which were discarded on this AAL5 VCC at the interface associated with an AAL5 entity because they were not fully re-assembled within the required time period. If the re-assembly timer is not supported, then this object contains a zero value.

5.13.4.11.5 aal5VccOverSizedSDUs

The number of AAL5 CPCS PDUs discarded on this AAL5 VCC at the interface associated with an AAL5 entity because the AAL5 SDUs were too large.

5.13.4.12 atmTrafficDescrParamIndexNext

OID:

- atmTrafficDescrParamIndexNext - { atmMIBObjects 13 } 1.3.6.1.2.1.10.37.1.13

This object contains an appropriate value to be used for atmTrafficDescrParamIndex when creating entries in the atmTrafficDescrParamTable. The value 0 indicates that no unassigned entries are available. To obtain the atmTrafficDescrParamIndex value for a new entry, the manager issues a management protocol retrieval operation to obtain the current value of this object. After each retrieval, the agent should modify the value to the next unassigned index. After a manager retrieves a value the agent will determine through its local policy when this index value is made available for reuse.

This object may be used in conjunction with the atmTrafficDescrParamTable for the creation of new table entries.

5.14 ATM-FORUM-SNMP-M4-MIB — AF-NM-0095.001

5.14.1 Introduction

This MIB describes objects for managing the ATM Forum's M4 interface. The AF-NM-0095.001 document used as a reference was dated July 1998. The LAST-UPDATED for the atmfM4MIB was "9805140000Z."

5.14.2 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprise - { private 4 } 1.3.6.1.4.4
- atmForum - { enterprises 353 } 1.3.6.1.4.4.353
- atmForumNetworkManagement - { atmForum 5 } 1.3.6.1.4.4.353.5
- atmfM4 - { atmForumNetworkManagement 1 } 1.3.6.1.4.4.353.5.1
- atmfM4SnmpNEView - { atmfM4 3 } 1.3.6.1.4.4.353.5.1.3
- atmfM4MIB - { atmfM4SnmpNEView 1 } 1.3.6.1.4.4.353.5.1.3.1
- atmfM4MIBObjects - { atmfM4MIB 1 } 1.3.6.1.4.4.353.5.1.3.1.1
- atmfM4TcAdapterTable - { atmfM4MIBObjects 7 } 1.3.6.1.4.4.353.5.1.3.1.1.7

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- transmission - { mib-2 10 } 1.3.6.1.2.1.10
- atmMIB - { mib-2 37 } 1.3.6.1.2.1.10.37
- atmMIBObjects - { atmMIB 1 } 1.3.6.1.2.1.10.37.1

5.14.3 MIB Objects

Only the atmfM4TcACellScrambling object which is part of the atmfM4TcAdapterTable is supported in this MIB.

The table below lists objects supported in this MIB. Objects that are not listed are not supported.

Table 136. ATM-FORUM-SNMP-M4-MIB Objects

Object	OID	Syntax	Access	Status
atmfM4TcAdaptorTable	{ atmfM4MIBObjects 7 }	SEQUENCE OF atmfM4TcAdaptorEntry	not-accessible	current
atmfMCellProtoCurrTable	{ pdnAtmM4ExtMIBObjects 15 }	SEQUENCE OF pdnAtmfM4CellProtoCurrEntry	not-accessible	current
atmfM4CellProtoHistTable	{ pdnAtmM4ExtMIBObjects 16 }	SEQUENCE OF pdnAtmfM4CellProtoHistEntry	not-accessible	current

5.14.3.1 atmfM4TcAdapterTable

Table to configure extensions for the TC Adapter. This table consists of a sequence of atmfM4AdapterEntry which contains the TC Adapter fields.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- atmfM4TcAdapterTable - { atmfM4MIBObjects 7 } 1.3.6.1.4.4.353.5.1.3.1.1.7
- atmfM4TcAdapterEntry - { atmfM4TcAdapterTable 1} 1.3.6.1.4.4.353.5.1.3.1.1.7.1

Table 137. atmfM4PhysPathTpTable

Object	OID	Syntax	Access	Status
atmfM4TcACellScrambling	{ atmfM4TcAdapterEntry 1 }	TruthValue	read-write	current

5.14.3.1.1 atmfM4TcACellScrambling

This object allows the ATM cell scrambling function to be enabled or disabled.

- true(1) - Cell scrambling is activated
- false(2) - Cell scrambling is not activated

This object defaults to true(1).

5.14.3.2 atmfM4CellProtoCurrTable

The ATM Forum M4 Cell Protocol Monitoring Current Data table. This table maintains per-interface statistics for the fifteen-minute interval currently being collected. Each ATM interface (UNI, BICI, BISSI) automatically has an entry in this table associated with it.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- atmfM4CellProtoCurrTable - { atmfM4MIBObjects 15 } 1.3.6.1.4.4.353.5.1.3.1.1.15
- atmfM4CellProtoCurrEntry - { atmfM4CellProtoCurrTable 1} 1.3.6.1.4.4.353.5.1.3.1.1.15.1

Table 138. atmfM4CellProtoCurrTable

Object	OID	Syntax	Access	Status
atmfM4CellProtoCurrSuspect	{ atmfM4CellProtoCurrEntry 1 }	TruthValue	read-only	current
atmfM4CellProtoCurrElapsedTime	{ atmfM4CellProtoCurrEntry 2 }	TimeInterval	read-only	current
atmfM4CellProtoCurrSupprIntvls	{ atmfM4CellProtoCurrEntry 3 }	Gauge32	read-only	current
atmfM4CellProtoCurrProtoErrors	{ atmfM4CellProtoCurrEntry 4 }	Gauge32	read-only	current
atmfM4CellProtoCurrInOAMCells	{ atmfM4CellProtoCurrEntry 5 }	Gauge32	read-only	current

5.14.3.2.1 atmfM4CellProtoCurrSuspect

If true, the statistics in this entry may be unreliable.

5.14.3.3 atmfM4CellProtoCurrElapsedTime

Amount of time, measured in units of 0.01 second, that statistics for this entry (the current interval) have been counted.

5.14.3.3.1 atmfM4CellProtoCurrSupprIntvls

This attribute is non-zero only if the ATM NE is suppressing ATM Cell Protocol Monitoring History Data entry creation when the current interval terminates with “all-zeroes” performance measurements.

When non-zero, this attribute represents the number of “all-zeroes” intervals that were suppressed immediately prior to the current interval.

Note that the suppression of “all-zeroes” intervals is controlled by the atmfM4NeSuppressZeroStats object.

5.14.3.3.2 atmfM4CellProtoCurrProtoErrors

The number of ATM cells dropped on this interface, due to an unrecognized field or set of fields in the ATM cell header, since the start of this interval.

5.14.3.3.3 atmfM4CellProtoCurrInOAMCells

The number of OAM cells received at this interface since the start of this interval.

5.14.3.4 atmfM4CellProtoHistTable

The ATM Forum M4 Cell Protocol Monitoring History Datatable.This table maintains per-interface statistics for previous fifteen-minute intervals. Each ATM interface (UNI, BICI, BISSI) automatically has an entry in this table associated with it for each fifteen-minute interval in which statistics are collected for it.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)
- atmfM4CellProtoHistIndex

OIDs:

- atmfM4CellProtoHistTable - { atmfM4MIBObjects 16 } 1.3.6.1.4.4.353.5.1.3.1.1.16
- atmfM4CellProtoHistEntry - { atmfM4CellProtoHistTable 1} 1.3.6.1.4.4.353.5.1.3.1.1.16.1

Table 139. atmfM4CellProtoHistTable

Object	OID	Syntax	Access	Status
atmfM4CellProtoHistIndex	{ atmfM4CellProtoHistEntry 1 }	INTEGER(1…96)	not-accessible	current
atmfM4CellProtoHistSuspect	{ atmfM4CellProtoHistEntry 2 }	TruthValue	read-only	current
atmfM4CellProtoHistElapsedTime	{ atmfM4CellProtoHistEntry 3 }	TimeInterval	read-only	current
atmfM4CellProtoHistSupprIntvls	{ atmfM4CellProtoHistEntry 4 }	Gauge32	read-only	current
atmfM4CellProtoHistProtoErrors	{ atmfM4CellProtoHistEntry 5 }	Gauge32	read-only	current
atmfM4CellProtoHistInOAMCells	{ atmfM4CellProtoHistEntry 6 }	Gauge32	read-only	current

5.14.3.4.1 atmfM4CellProtoHistIndex

A number between 1 and 96, which identifies the interval for which the set of statistics in this entry was collected.

The interval identified by 1 is the most recently completed 15 minute interval, and the interval identified by N is the interval immediately preceding the one identified by N-1.

5.14.3.4.2 atmfM4CellProtoHistSuspect

If true, the statistics in this entry may be unreliable.

5.14.3.4.3 atmfM4CellProtoHistElapsedTime

Amount of time, measured in units of 0.01 second, that statistics for this entry (the current interval) have been counted.

5.14.3.4.4 atmfM4CellProtoHistSupprIntvls

This attribute is non-zero only if the ATM NE is suppressing ATM Cell Protocol Monitoring History Data entry creation when the current interval terminates with “all-zeroes” performance measurements.

When non-zero, this attribute represents the number of “all-zeroes” intervals that were suppressed immediately prior to this “non-all-zeroes” history interval.

Note that the suppression of “all-zeroes” intervals is controlled by the atmfM4NeSuppressZeroStats object.

5.14.3.4.5 atmfM4CellProtoHistProtoErrors

The number of ATM cells dropped on this interface, due to an unrecognized field or set of fields in the ATM cell header, during this interval.

5.14.3.4.6 atmfM4CellProtoHistInOAMCells

The number of OAM cells received at this interface during this interval.

5.15 BRIDGE-MIB — RFC 1493

5.15.1 Introduction

This MIB defines objects for managing MAC bridges based on the IEEE 802.1D standard between Local Area Network (LAN) segments. The RFC used as a reference was dated July 1993.

5.15.2 Textual Conventions

The following definitions are made in this RFC:

- MacAddress - A 6 octet address in the “canonical” order defined by IEEE 802.1a, that is, as it were transmitted least significant bit first
  — SYNTAX OCTET STRING (SIZE (6))
- BridgeId - The Bridge-Identifier as used in the Spanning Tree Protocol to uniquely identify a bridge.
  — SYNTAX OCTET STRING (SIZE (8))
- Timeout - A Spanning Tree Protocol timer in units of 1/100 seconds.
  — INTEGER

5.15.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- dot1dBridge - { mib-2 17 } 1.3.6.1.2.1.17
- dot1dBase - { dot1dBridge 1 } 1.3.6.1.2.1.17.1
- dot1dStp - { dot1dBridge 2 } 1.3.6.1.2.1.17.2
- dot1dSr - { dot1dBridge 3 } 1.3.6.1.2.1.17.3
- dot1dTp - { dot1dBridge 4 } 1.3.6.1.2.1.17.4
- dot1dStatic - { dot1dBridge 5 } 1.3.6.1.2.1.17.5

5.15.4 MIB Objects

The following groups are defined in this MIB:

Table 140. BRIDGE-MIB Objects

Groups	OID	Description	Supported
dot1dBase	{ dot1dBridge 1 }	This group contains objects which are applicable to all types of bridges.	Y
dot1dStp	{ dot1dBridge 2 }	This group contains the objects that denote the bridge’s state with respect to the Spanning Tree Protocol.	N
dot1dSr	{ dot1dBridge 3 }	This group contains the objects that describe the entity’s state with respect to source route bridging.	N
dot1dTp	{ dot1dBridge 4 }	This group contains objects that describe the entity’s state with respect to transparent bridging.	Y
dot1dStatic	{ dot1dBridge 5 }	This group contains objects that describe the entity’s state with respect to destination-address filtering.	Y

5.15.4.1 dot1dBase Group

OID:

- dot1dBase - { dot1dBridge 1 } 1.3.6.1.2.1.17.1

Table 141. dot1dBase Group

Object	OID	Syntax	Access	Status	Supported
dot1dBaseBridgeAddress	{ dot1dBase 1 }	MacAddress	read-only	mandatory	N
dot1dBaseNumPorts	{ dot1dBase 2 }	INTEGER	read-only	mandatory	Y
dot1dBaseType	{ dot1dBase 3 }	INTEGER unknown (1) transparent-only (2) sourceroute-only (3) srt (4)	read-only	mandatory	Y
dot1dBasePortTable	{ dot1dBase 4 }	SEQUENCE of dot1dBasePortEntry	not-accessible	mandatory	Y

5.15.4.1.1 dot1dBaseBridgeAddress

The MAC address used by this bridge when it must be referred to in a unique fashion. It is recommended that this be the numerically smallest MAC address of all ports that belong to this bridge. However it is only required to be unique. When concatenated with dot1dStpPriority a unique BridgeIdentifier is formed which is used in the Spanning tree protocol.

Valid Object Value(s)

This object will not be supported as there is no MAC address for a bridge in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

5.15.4.1.2 dot1dBaseNumPorts

The number of ports controlled by this bridging entity.

Valid Object Value(s)

Each chassis has the following bridge ports:

- Ethernet Port(s) - 3
- ADSL Ports - 384
  Each DSL Port has 8 connections to the Bridge.
  (48 DSL Ports * 8 Connections to the Bridge = 384 Bridge Ports)
- Management Ports - 2

Which gives a total of 389 bridge ports.

5.15.4.1.3 dot1dBaseType

Indicates what type of bridging this bridge can perform. If a bridge is actually performing a certain type of bridging this is indicated by entries in the port table for the given type.

- unknown(1)
- transparent-only(2)
- soureceroute-only(3)
- srt(4)

Valid Object Value(s)

This object will always return the transparent-only(2) value.

5.15.4.1.4 dot1dBasePortTable

This is a generic bridge port table that contains generic information about every port that is associated with this bridge. Transparent, source-route, and srt ports are included.

Index:

- dot1dBasePort

OIDs:

- dot1dBasePortTable - { dot1dBase 4 } 1.3.6.1.2.1.17.1.4
- dot1dBasePortEntry - { dot1dBasePortTable 1 } 1.3.6.1.2.1.17.1.4.1

Table 142. dot1dBasePortTable

Object	OID	Syntax	Access	Status	Supported
dot1dBasePort	{ dot1dBasePortEntry 1 }	INTEGER (1…65535)	read-only	mandatory	Y
dot1dBasePortIfIndex	{ dot1dBasePortEntry 2 }	INTEGER	read-only	mandatory	Y
dot1dBasePortCircuit	{ dot1dBasePortEntry 3 }	OBJECT IDENTIFIER	read-only	mandatory	Y
dot1dBasePortDelayExceededDiscards	{ dot1dBasePortEntry 4 }	Counter	read-only	mandatory	N
dot1dBaseMtuExceededDiscards	{ dot1dBasePortEntry 5 }	Counter	read-only	mandatory	N

5.15.4.1.4.1 dot1dBasePort

The port number of the port for which this entry contains bridge management information.

5.15.4.1.4.2 dot1dBasePortIfIndex

The value of this instance of the ifIndex (Section 5.3.4.3.1, Page 37) object, defined in MIB-II, for the interface corresponding to this port.

5.15.4.1.4.3 dot1dBasePortCircuit

For a port which (potentially) has the same value of dot1dBasePortIfIndex as another port on the same bridge, this object contains the name of an object instance unique to this port. For example, in the case where multiple ports co respond one- to-one with multiple X.25 virtual circuits, this value might identify an (for example, the first) object instance associated with the X.25 virtual circuit corresponding to this port.

For a port which has a unique value of dot1dBasePortIfIndex, this object can have the value { 0 0 }.

Valid Object Value(s)

Bridge ports do not necessarily have the same ifIndex. For example, for each DSL Port, there a 8 Bridge Ports (1 per Priority Inter-Working Group). As a result, the following are defined for this object:

Company Enterprise.[a]	1.14.17.7	ips-dot1dBasePortCircuit
	1.14.17.7.1	ips-piwg-1
	1.14.17.7.2	ips-piwg-2
	1.14.17.7.3	ips-piwg-3
	1.14.17.7.4	ips-piwg-4
	1.14.17.7.5	ips-piwg-5
	1.14.17.7.6	ips-piwg-6
	1.14.17.7.7	ips-piwg-7
	1.14.17.7.8	ips-piwg-8

a. Refer to Section 5.2.3.1.2, Page 22, for a definition of *Company Enterprise*.

dot1dBasePortCircuit Values

5.15.4.1.4.4 dot1dBasePortDelayExceededDiscards

The number of frames discarded by this port due to excessive transit delay through the bridge. It is incremented by both transparent and source route bridges.

5.15.4.1.4.5 dot1dBasePortMtuExceededDiscards

The number of frames discarded by this port due to an excessive size. It is incremented by both transparent and source route bridges.

5.15.4.2 dot1dStp Group

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM do not support the Spanning Tree Protocol so this group is not supported.

5.15.4.3 dot1dSr Group

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM do not support source route bridging so this group is not supported.

5.15.4.4 dot1dTp Group

OID:

- dot1dTp - { dot1dBridge 4 } 1.3.6.1.2.1.17.4

Table 143. dot1dTp Group

Object	OID	Syntax	Access	Status	Supported
dot1dTpLearnedEntityDiscards	{ dot1dTp 1 }	Counter	read-only	mandatory	N
dot1dTpAgingTime	{ dot1dTp 2 }	INTEGER (10…1000000)	read-write	mandatory	Y
dot1dTpFdbTable	{ dot1dTp 3 }	SEQUENCE of dot1dTpFdbEntry	not-accessible	mandatory	Y
dot1dTpPortTable	{ dot1dTp 4 }	SEQUENCE of dot1dTpPortEntry	not-accessible	mandatory	Y

5.15.4.4.1 dot1dTpLearnedEntryDiscards

The total number of Forwarding Database entries, which have been or would have been learnt, but have been discarded due to a lack of space to store them in the Forwarding Database. If this counter is increasing, it indicates that the Forwarding Database is regularly becoming full (a condition which has unpleasant performance effects on the subnetwork). If this counter has a significant value but is not presently increasing, it indicates that the problem has been occurring but is not persistent.

5.15.4.4.2 dot1dTpAgingTime

The timeout period in seconds for aging out dynamically learned forwarding information. 802.1D-1990 recommends a default of 300 seconds.

This object will default to 300 seconds.

5.15.4.4.3 dot1dTpFdbTable

The Forwarding Database for Transparent Bridges is a table that contains information about unicast entries for which the bridge has forwarding and/or filtering information. This information is used by the transparent bridging function in determining how to propagate a received frame.

Each entry contains information about a specific unicast MAC address for which the bridge has some forwarding and/or filtering information.

This table reports MAC addresses learned on each port, regardless of which Filtering Database they have been learnt in. If an address has been learnt in multiple databases on a single port, report it only once. If an address has been learnt in multiple databases on more than one port, report the entry on any one of the valid ports.

Index:

- dot1dTpFdbAddress

OIDs:

- dot1dTpFdbTable - { dot1dTp 3 } 1.3.6.1.2.1.17.4.3
- dot1dTpFdbEntry - { dot1dTpFdbTable 1 } 1.3.6.1.2.1.17.4.3.1

Table 144. dot1dTpFdbTable

Object	OID	Syntax	Access	Status	Supported
dot1dTpFdbAddress	{ dot1dTpFdbEntry 1 }	MacAddress	read-only	mandatory	Y
dot1dTpFdbPort	{ dot1dTpFdbEntry 2 }	INTEGER	read-only	mandatory	Y
dot1dTpFdbStatus	{ dot1dTpFdbEntry 3 }	INTEGER other (1) invalid (2) learned (3) self (4) mgmt (5)	read-only	mandatory	Y

5.15.4.4.3.1 dot1dTpFdbAddress

A unicast MAC address for which the bridge has forwarding and/or filtering information.

5.15.4.4.3.2 dot1dTpFdbPort

Either the value “0,” or the port number of the port on which a frame having a source address equal to the value of the corresponding instance of dot1dTpFdbAddress has been seen. A value of “0” indicates that the port number has not been learned but that the bridge does have some forwarding/filtering information about this address (for example, in

the dot1dStaticTable). Implementors are encouraged to assign the port value to this object whenever it is learned even for addresses for which the corresponding value of dot1dTpFdbStatus is not learned(3).

5.15.4.4.3.3 dot1dTpFdbStatus

The status of this entry in the table. The meanings of the values are:

- other(1) - none of the following. This would include the case where some other MIB object (not the corresponding instance of dot1dTpFdbPort, nor an entry in the dot1dStaticTable) is being used to determine if and how frames addressed to the value of the corresponding instance of dot1dTpFdbAddress are being forwarded.
- invalid(2) - this entry is not longer valid (for example, it was learned but has since aged-out), but has not yet been flushed from the table.
- learned(3) - the value of the corresponding instance of dot1dTpFdbPort was learned, and is being used.
- self(4) - the value of the corresponding instance of dot1dTpFdbAddress represents one of the bridge's addresses. The corresponding instance of dot1dTpFdbPort indicates which of the bridge's ports has this address.
- mgmt(5) - the value of the corresponding instance of dot1dTpFdbAddress is also the value of an existing instance of dot1dStaticAddress.

5.15.4.4.4 dot1dTpPortTable

The Port Table for Transparent Bridges is table that contains information about every port that is associated with this transparent bridge.

Each entry is list of information for each port of a transparent bridge. Also, this table is port-based and is not affected by multiple Filtering Databases or multiple VLANs. The counters should include frames received or transmitted for all VLANs. Note that equivalent 64-bit port statistics counters, as well as other objects to represent the upper 32 bits of these counters, are defined in this document for high capacity network interfaces.

Index:

- dot1dTpPort

OIDs:

- dot1dTpPortTable - { dot1dTp 4 } 1.3.6.1.2.1.17.4.4
- dot1dTpPortEntry - { dot1dTpPortTable 1 } 1.3.6.1.2.1.17.4.4.1

Table 145. dot1dTpPortTable

Object	OID	Syntax	Access	Status	Supported
dot1dTpPort	{ dot1dTpPortEntry 1 }	INTEGER	read-only	mandatory	Y
dot1dTpPortMaxInfo	{ dot1dTpPortEntry 2 }	INTEGER	read-only	mandatory	Y
dot1dTpPortInFrames	{ dot1dTpPortEntry 3 }	Counter	read-only	mandatory	Y
dot1dTpPortOutFrames	{ dot1dTpPortEntry 4 }	Counter	read-only	mandatory	Y
dot1dTpPortInDiscards	{ dot1dTpPortEntry 5 }	Counter	read-only	mandatory	Y

5.15.4.4.4.1 dot1dTpPort

The port number of the port for which this entry information.

5.15.4.4.4.2 dot1dTpPortMaxInfo

The maximum size of the INFO (non-MAC) field that this port will receive or transmit.

Valid Object Value(s)

This object will return a 1500.

5.15.4.4.4.3 dot1dTpPortInFrames

The number of frames that have been received by this port from its segment. Note that a frame received on the interface corresponding to this port is only counted by this object if and only if it is for a protocol being processed by the local bridging function, including bridge management frames.

5.15.4.4.4.4 dot1dTpPortOutFrames

The number of frames that have been transmitted by this port to its segment. Note that a frame transmitted on the interface corresponding to this port is only counted by this object if and only if it is for a protocol being processed by the local bridging function, including bridge management frames.

5.15.4.4.4.5 dot1dTpPortInDiscards

Count of valid frames received which were discarded (that is, filtered) by the Forwarding Process.

5.15.4.5 dot1dStatic Group

The Static (Destination-Address Filtering) Database Group if fully supported. The following sections provide clarification for objects contained in this group.

OID:

- dot1dStatic - { dot1dBridge 5 } 1.3.6.1.2.1.17.5

Table 146. dot1dStatic Group

Object	OID	Syntax	Access	Status	Supported
dot1dStaticTable	{ dot1dStatic 1 }	SEQUENCE of dot1dStaticEntry	not-accessible	mandatory	Y

5.15.4.5.1 dot1dStaticTable

A table containing filtering information configured into the bridge by (local or network) management specifying the set of ports to which frames received from specific ports and containing specific destination addresses are allowed to be forwarded. The value of zero in this table as the port number from which frames with a specific destination address are received, is used to specify all ports for which there is no specific entry in this table for that particular destination address. Entries are valid for unicast and for group/broadcast addresses.

Each entry consists of filtering information configured into the bridge by (local or network) management specifying the set of ports to which frames received from a specific port and containing a specific destination address are allowed to be forwarded. Entries read from this table include all static entries from all of the Filtering Databases. Entries for the same MAC address and receive port in more than one Filtering Database must appear only once since these are the indices of this table. This table should be implemented as read-only in devices that support multiple Forwarding Databases - instead, write access should be provided through dot1qStaticUnicastTable and dot1qStaticMulticastTable, as defined in this document.

Indices:

- dot1dStaticAddress
- dot1dStaticReceivePort

OIDs:

- dot1dStaticTable - { dot1dStatic 1 } 1.3.6.1.2.1.17.5.1
- dot1dStaticEntry - { dot1dStaticTable 1 } 1.3.6.1.2.1.17.5.1.1

Table 147. dot1dStaticTable

Object	OID	Syntax	Access[a]	Status	Supported
dot1dStaticAddress	{ dot1dStaticEntry 1 }	MacAddress	read-only	mandatory	Y
dot1dStaticReceivePort	{ dot1dStaticEntry 2 }	INTEGER	read-only	mandatory	Y
dot1dStaticAllowedToGoTo	{ dot1dStaticEntry 3 }	OCTET STRING	read-only	mandatory	Y
dot1dStaticStatus	{ dot1dStaticEntry 4 }	INTEGER other (1) invalid (2) permanent (3) deleteOnReset (4) deleteOnTimeout (5)	read-only	mandatory	Y

a. Access restricted to read-only per RFC 2674, dot1dStaticTable section, as captured in paragraph above.

5.15.4.5.1.1 dot1dStaticAddress

The destination MAC address in a frame to which this entry's filtering information applies. This object can take the value of a unicast address, a group address or the broadcast address.

5.15.4.5.1.2 dot1dStaticReceivePort

Either the value “0,” or the port number of the port from which a frame must be received in order for this entry's filtering information to apply. A value of zero indicates that this entry applies on all ports of the bridge for which there is no other applicable entry.

5.15.4.5.1.3 dot1dStaticAllowedToGoTo

The set of ports to which frames received from a specific port and destined for a specific MAC address, are allowed to be forwarded. Each octet within the value of this object specifies a set of eight ports, with the first octet specifying ports 1 through 8, the second octet specifying ports 9 through 16, etc. Within each octet, the most significant bit represents the lowest numbered port, and the least significant bit represents the highest numbered port. Thus, each port of the bridge is represented by a single bit within the value of this object. If that bit has a value of “1” then that port is included in the set of ports; the port is not included if its bit has a value of “0.” (Note that the setting of the bit corresponding to the port from which a frame is received is irrelevant.) The default value of this object is a string of ones of appropriate received is irrelevant.) The default value of this object is a string of ones of appropriate length.

5.15.4.5.1.4 dot1dStaticStatus

This object indicates the status of this entry. The default value is permanent(3).

- other(1) - this entry is currently in use but the conditions under which it will remain so are different from each of the following values.
- invalid(2) - writing this value to the object removes the corresponding entry.
- permanent(3) - this entry is currently in use and will remain so after the next reset of the bridge.
- deleteOnReset(4) - this entry is currently in use and will remain so until the next reset of the bridge.
- deleteOnTimeout(5) - this entry is currently in use and will remain so until it is aged out.

5.16 P-BRIDGE-MIB — RFC 2674

5.16.1 Introduction

The Bridge MIB Extension module for managing Priority and Multicast Filtering, defined by IEEE 802.1D-1998. The LAST-UPDATED for this MIB in the RFC was “9908250000Z.”

5.16.2 Textual Conventions

The following textual conventions are made in this RFC:

- EnabledStatus - A simple status value for an object.
  - — INTEGER
    - enabled(1)
    - disabled(2)

5.16.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- dot1dBridge - { mib-2 17 } 1.3.6.1.2.1.17
- pBridgeMIB - { dot1dBridge 6 } 1.3.6.1.2.1.17.6
- pBridgeMIBObjects - { pBridgeMIB 1 } 1.3.6.1.2.1.17.6.1
- dot1dExtBase - { pBridgeMIBObjects 1 } 1.3.6.1.2.1.17.6.1.1
- dot1dPriority - { pBridgeMIBObjects 2 } 1.3.6.1.2.1.17.6.1.2
- dot1dGarp - { pBridgeMIBObjects 3 } 1.3.6.1.2.1.17.6.1.3
- dot1dGmrp - { pBridgeMIBObjects 4 } 1.3.6.1.2.1.17.6.1.4

5.16.4 MIB Objects

Table 148. P-BRIDGE-MIB Objects (1 of 2)

Object	OID	Description	Supported
dot1dExtBase	{ pBridgeMIBObjects 1 }	Bridge MIB Extensions Base Group This group contains the objects which are applicable to all bridges implementing the traffic class and multicast filtering features of IEEE 802.1D-1998 [802.1D]. It includes per device configuration of GARP and GMRP protocols	Y
dot1dPriority	{ pBridgeMIBObjects 2 }	Bridge Priority Group This group contains the objects for configuring and reporting status of priority-based queuing mechanisms in a bridge. This includes per-port user_priority treatment, mapping of user_priority in frames into internal traffic classes and outbound user_priority and access priority.	Y
dot1dGarp	{ pBridgeMIBObjects 3 }	Generic Attribute Registration Protocol (GARP) Group This group contains the objects for configuring and reporting on operations of the Generic Attribute Registration Protocol (GARP).	N

Table 148. P-BRIDGE-MIB Objects (2 of 2)

Object	OID	Description	Supported
dot1dGmrp	{ pBridgeMIBObjects 4 }	GARP Multicast Registration Protocol (GMRP) This group contains the objects for configuring and reporting on operation of the GARP Multicast Registration Protocol (GMRP).	N
dot1dTp	{ dot1dBridge 4 }	Extension of the Transparent Bridges This group contains objects that describe the entity's state with respect to high capacity transparent bridging. This group consists primarily the High Capacity Port Table for Transparent Bridges.	N

5.16.4.1 dot1dExtBase Group

This group contains the objects which are applicable to all bridges implementing the traffic class and multicast filtering features of IEEE 802.1D-1998 [802.1D]. It includes per device configuration of GARP and GMRP protocols.

OID:

- dot1dExtBase - { pBridgeMIBObjects 1 } 1.3.6.1.2.1.17.6.1.1

Table 149. dot1dExtBase Group

Object	OID	Syntax	Access	Status	Supported
dot1dDeviceCapabilities	{ dot1dExtBase 1 }	Bits dot1dExtendedFilteringServices(0) dot1dTrafficClasses(1) dot1qStaticEntryIndividualPort(2) dot1qIVLCapable(3) dot1qSVLCapable(4) dot1qHybridCapable(5) dot1qConfigurablePvidTagging(6) dot1dLocalVlanCapable(7)	read-only	current	Y
dot1dTrafficClassesEnabled	{ dot1dExtBase 2 }	TruthValue	read-write	current	Y
dot1dGmrpStatus	{ dot1dExtBase 3 }	EnabledStatus	read-only[a]	current	Y
dot1dPortCapabilitiesTable	{ dot1dExtBase 4 }	SEQUENCE of dot1dPortCapabilitiesEntry	not-accessible	current	Y

a. Product imposed limit.

5.16.4.1.1 dot1dDeviceCapabilities

Indicates the optional parts of IEEE 802.1D and 802.1Q that are implemented by this device and are manageable through this MIB. Capabilities that are allowed on a per-port basis are indicated in dot1dPortCapabilities.

The following are definitions for this objects' syntax:

- dot1dExtendedFilteringServices(0) - can perform filtering of individual multicast addresses controlled by GMRP.
- dot1dTrafficClasses(1) - can map user priority to multiple traffic classes.
- dot1qStaticEntryIndividualPort(2) - dot1qStaticUnicastReceivePort (Section 5.17.4.3.1.2, Page 288) & dot1qStaticMulticastReceivePort (Section 5.17.4.3.2.2, Page 289) can represent non-zero entries.

- dot1qIVLCapable(3) - Independent VLAN Learning.
- dot1qSVLCapable(4) - Shared VLAN Learning.
- dot1qHybridCapable(5) - both IVL & SVL simultaneously.
- dot1qConfigurablePvidTagging(6) - whether the implementation supports the ability to override the default PVID setting and its egress status (VLAN-Tagged or Untagged) on each port.
- dot1dLocalVlanCapable(7) - can support multiple local bridges, outside of the scope of 802.1Q defined VLANs.

Valid Object Value(s)

The following bits should be set:

- dot1dTrafficClasses(1)
- dot1qIVLCapable(3)
- dot1qConfigurablePvidTAgging(6)

5.16.4.1.2 dot1dTrafficClassesEnabled

The value true(1) indicates that Traffic Classes are enabled on this bridge. When false(2), the bridge operates with a single priority level for all traffic. The default value is true.

Valid Object Value(s)

- true(1)

5.16.4.1.3 dot1dGmrpStatus

The administrative status requested by management for GMRP. The value enabled(1) indicates that GMRP should be enabled on this device, in all VLANs, on all ports for which it has not been specifically disabled. When disabled(2), GMRP is disabled, in all VLANs, on all ports and all GMRP packets are forwarded transparently. This object affects both Applicant and Registrar state machines. A transition from disabled(2) to enabled(1) will cause a reset of all GMRP state machines on all ports. The default value is enabled.

Valid Object Value(s)

- disabled(2)

5.16.4.1.4 dot1dPortCapabilitiesTable

This object is a table that contains capabilities information about every port that is associated with this bridge.

Augments:

- dot1dBasePortTable (Section 5.15.4.1.4, Page 259)

Index:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1dPortCapabilitiesTable - { dot1dExtBase 4 } 1.3.6.1.2.1.17.6.1.1.4
- dot1dPortCapabilitiesEntry - { dot1dPortCapabilitiesTable 1 } 1.3.6.1.2.1.17.6.1.1.4.1

Table 150. dot1dPortCapabilitiesTable

Object	OID	Syntax	Access	Status	Supported
dot1dPortCapabilities	{ dot1dPortCapabilitiesEntry 1 }	Bits dot1qDot1qTagging(0) dot1qConfigurableAcceptableFrameTypes(1) dot1qIngressFiltering(2) dot1qIngressFiltering(2)	read-only	current	Y

5.16.4.1.4.1 dot1dPortCapabilities

Indicates the parts of IEEE 802.1D and 802.1Q that are optional on a per-port basis that are implemented by this device and are manageable through this MIB.

The following are definitions for this objects' syntax:

- dot1qDot1qTagging(0) - supports 802.1Q VLAN tagging of frames and GVRP.
- dot1qConfigurableAcceptableFrameTypes(1) - allows modified values of dot1qPortAcceptableFrameTypes (Section 5.17.4.4.5.2, Page 294).
- dot1qIngressFiltering(2) - supports the discarding of any frame received on a Port whose VLAN classification does not include that Port in its Member set.

Valid Object Value(s)

The following bits should be set:

- dot1qConfigurableAcceptableFrameTypes(1)
- dot1qIngressFiltering(2)

5.16.4.2 dot1dPriority Group

This group contains the objects for configuring and reporting status of priority-based queuing mechanisms in a bridge. This includes per-port user_priority treatment, mapping of user_priority in frames into internal traffic classes and outbound user_priority and access priority.

OID:

- dot1dPriority - { pBridgeMIBObjects 2 } 1.3.6.1.2.1.17.6.1.2

Table 151. dot1dPriority Group

Object	OID	Syntax	Access	Status	Supported
dot1dPortPriorityTable	{ dot1dPriority 1 }	SEQUENCE of dot1dPortPriorityEntry	not-accessible	current	Y
dot1dUserPriorityRegenTable	{ dot1dPriority 2	SEQUENCE of dot1dUserPriorityRegenEntry	not-accessible	current	N
dot1dTrafficClassTable	{ dot1dPriority 3 }	SEQUENCE of dot1dTrafficClassEntry	not-accessible	current	Y

Table 151. dot1dPriority Group

Object	OID	Syntax	Access	Status	Supported
dot1dPortOutboundAccessPriorityTable	{ dot1dPriority 4 }	SEQUENCE of dot1dPortOutboundAccessPriorityEntry	not-accessible	current	N

5.16.4.2.1 dot1dPortPriorityTable

The Port Priority Table contains information about every port that is associated with this transparent bridge. Each entry consists of a list of Default User Priorities for each port of a transparent bridge.

Augments:

- dot1dBasePortTable (Section 5.15.4.1.4, Page 259)

Index:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1dPortPriorityTable - { dot1dPriority 1 } 1.3.6.1.2.1.17.6.1.2.1
- dot1dPortPriorityEntry - { dot1dPortPriorityTable 1 } 1.3.6.1.2.1.17.6.1.2.1.1

Table 152. dot1dPortPriorityTable

Object	OID	Syntax	Access	Status	Supported
dot1dPortDefaultUserPriority	{ dot1dPortPriorityEntry 1 }	INTEGER (0…7)	read-write	current	Y
dot1dPortNumTrafficClasses	{ dot1dPortPriorityEntry 2 }	INTEGER (1…8)	read-only[a]	current	Y

a. Product imposed restriction.

5.16.4.2.1.1 dot1dPortDefaultUserPriority

The default ingress User Priority for this port. This only has effect on media, such as Ethernet, that do not support native User Priority.

5.16.4.2.1.2 dot1dPortNumTrafficClasses

The number of egress traffic classes supported on this port. This object may optionally be read-only.

Valid Object Value(s)

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM support four egress traffic classes.

5.16.4.2.2 dot1dUserPriorityRegenTable

The User Priority Regeneration Table is a list of Regenerated User Priorities for each received User Priority on each port of a bridge. The Regenerated User Priority value may be used to index the Traffic Class Table for each input port. This only has effect on media that support native User Priority. The default values for Regenerated User Priorities are the same as the User Priorities. Each entry consists of a mapping of incoming User Priority to a Regenerated User Priority.

Indices:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)
- dot1dUserPriority

OIDs:

- dot1dUserPriorityRegenTable - { dot1dPriority 2 } 1.3.6.1.2.1.17.6.1.2.2
- dot1dUserPriorityRegenEntry - { dot1dUserPriorityRegenTable 1 } 1.3.6.1.2.1.17.6.1.2.2.1

Table 153. dot1dUserPriorityRegenTable

Object	OID	Syntax	Access	Status	Supported
dot1dUserPriority	{ dot1dUserPriorityRegenEntry 1 }	INTEGER (0…7)	not-accessible	current	N
dot1dRegenUserPriority	{ dot1dUserPriorityRegenEntry 2 }	INTEGER (0…7)	read-write	current	N

5.16.4.2.2.1 dot1dUserPriority

The User Priority for a frame received on this port.

5.16.4.2.2.2 dot1dRegenUserPriority

The Regenerated User Priority the incoming User Priority is mapped to for this port.

5.16.4.2.3 dot1dTrafficClassTable

The Traffic Class Table is a table mapping evaluated User Priority to Traffic Class, for forwarding by the bridge. Traffic class is a number in the range (0…(dot1dPortNumTrafficClasses [Section 5.16.4.2.1.2, Page 269]-1)).

Indices:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)
- dot1dTrafficClassPriority

OIDs:

- dot1dTrafficClassTable - { dot1dPriority 3 } 1.3.6.1.2.1.17.6.1.2.3
- dot1dTrafficClassEntry - { dot1dTrafficClassTable 1 } 1.3.6.1.2.1.17.6.1.2.3.1

Table 154. dot1dTrafficClassTable

Object	OID	Syntax	Access	Status	Supported
dot1dTrafficClassPriority	{ dot1dTrafficClassEntry 1 }	INTEGER (0…7)	not-accessible	current	Y
dot1dTrafficClass	{ dot1dTrafficClassEntry 2 }	INTEGER (0…7)	read-write	current	Y

5.16.4.2.3.1 dot1dTrafficClassPriority

The Priority value determined for the received frame. This value is equivalent to the priority indicated in

the tagged frame received, or one of the evaluated priorities, determined according to the media-type.

For untagged frames received from Ethernet media, this value is equal to the dot1dPortDefaultUserPriority (Section 5.16.4.2.1.1, Page 269) value for the ingress port.

For untagged frames received from non-Ethernet media, this value is equal to the dot1dRegenUserPriority (Section 5.16.4.2.2.2, Page 270) value for the ingress port and media-specific user priority.

5.16.4.2.3.2 dot1dTrafficClass

The Traffic Class the received frame is mapped to.

Valid Object Value(s)

The table below maps User Priority (dot1dTrafficClassPriority) to Traffic Class (dot1dTrafficClass) default. This table is extracted from the IEEE 802.1D Standard's *Recommended user priority to traffic class mappings* table. The default mapping is determined based on the product's support of 4 output classes (queues).

Table 155. Default dot1dTrafficClass Mapping

		Number of Available Traffic Classes							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
User Priority	**0 (Default)**	0	0	0	1	1	1	1	2
	1	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	1
	3	0	0	0	1	1	2	2	3
	4	0	1	1	2	2	3	3	4
	5	0	1	1	2	3	4	4	5
	6	0	1	2	3	4	5	5	6
	7	0	1	2	3	4	5	6	7

5.16.4.2.4 dot1dPortOutboundAccessPriorityTable

The Outbound Access Priority Table is a mapping Regenerated User Priority to Outbound Access Priority. This is a fixed mapping for all port types, with two options for 802.5 Token Ring. Each entry consists of regenerated User Priority to Outbound Access Priority mapping.

Indices:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)
- dot1dRegenUserPriority

OIDs:

- dot1dPortOutboundAccessPriorityTable - { dot1dPriority 4 } 1.3.6.1.2.1.17.6.1.2.4
- dot1dPortOutboundAccessPriorityEntry - { dot1dPortOutboundAccessPriorityTable 1 } 1.3.6.1.2.1.17.6.1.2.4.1

Table 156. dot1dPortOutboundAccessPriorityTable

Object	OID	Syntax	Access	Status	Supported
dot1dPortOutboundAccessPriority	{ dot1dPortOutboundAccessPriorityEntry 1 }	INTEGER (0…7)	read-only	current	N

5.16.4.2.4.1 dot1dPortOutboundAccessPriority

The Outbound Access Priority the received frame is mapped to.

5.16.4.3 dot1dGarp Group

This group contains the objects for configuring and reporting on operations of the Generic Attribute Registration Protocol (GARP).

OID:

- dot1dGarp - { pBridgeMIBObjects 3 } 1.3.6.1.2.1.17.6.1.3

Table 157. dot1dGarp Group

Object	OID	Syntax	Access	Status	Supported
dot1dPortGarpTable	{ dot1dGarp 1 }	SEQUENCE of dot1dPortGarpEntry	not-accessible	current	N

5.16.4.3.1 dot1dPortGarpTable

A table of GARP control information about every bridge port.

Augments:

- dot1dBasePortTable (Section 5.15.4.1.4, Page 259)

Index:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1dPortGarpTable - { dot1dGarp 1 } 1.3.6.1.2.1.17.6.1.3.1
- dot1dPortGarpEntry - { dot1dPortGarpTable 1 } 1.3.6.1.2.1.17.6.1.3.1.1

Table 158. dot1dPortGarpTable

Object	OID	Syntax	Access	Status	Supported
dot1dPortGarpJoinTime	{ dot1dPortGarpEntry 1 }	TimeInterval	read-write	current	N
dot1dPortGarpLeaveTime	{ dot1dPortGarpEntry 2 }	TimeInterval	read-write	current	N
dot1dPortGarpLeaveAllTime	{ dot1dPortGarpEntry 3 }	TimeInterval	read-write	current	N

5.16.4.3.1.1 dot1dPortGarpJoinTime

The GARP Join time, in centiseconds.

DEFVAL-20

5.16.4.3.1.2 dot1dPortGarpLeaveTime

The GARP Leave time, in centiseconds.

DEFVAL-60

5.16.4.3.1.3 dot1dPortGarpLeaveAllTime

The GARP LeaveAll time, in centiseconds.

DEFVAL-1000

5.16.4.4 dot1dGmrp Group

This group contains the objects for configuring and reporting on operations of the Generic Attribute Registration Protocol (GARP).

OID:

- dot1dGmrp - { pBridgeMIBObjects 4 } 1.3.6.1.2.1.17.6.1.4

Table 159. dot1dGmrp Group

Object	OID	Syntax	Access	Status	Supported
dot1dPortGmrpTable	{ dot1dGmrp 1 }	SEQUENCE of dot1dPortGmrpEntry	not-accessible	current	N

5.16.4.4.1 dot1dPortGmrpTable

A table of GMRP control and status information about every bridge port. Augments the dot1dBasePortTable.

Augments:

- dot1dBasePortTable (Section 5.15.4.1.4, Page 259)

Index:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1dPortGmrpTable - { dot1dGmrp 1 } 1.3.6.1.2.1.17.6.1.4.1
- dot1dPortGmrpEntry - { dot1dPortGmrpTable 1 } 1.3.6.1.2.1.17.6.1.4.1.1

Table 160. dot1dPortGmrpTable

Object	OID	Syntax	Access	Status	Supported
dot1dPortGmrpStatus	{ dot1dPortGmrpEntry 1 }	EnabledStatus	read-write	current	N
dot1dPortGmrpFailedRegistrations	{ dot1dPortGmrpEntry 2 }	Counter32	read-only	current	N
dot1dPortGmrpLastPduOrigin	{ dot1dPortGmrpEntry 3 }	MacAddress	read-only	current	N

5.16.4.4.1.1 dot1dPortGmrpStatus

The administrative state of GMRP operation on this port. The value enabled(1) indicates that GMRP is enabled on this port in all VLANs as long as dot1dGmrpStatus is also enabled(1). A value of disabled(2) indicates that GMRP is disabled on this port in all VLANs: any GMRP packets received are silently discarded and no GMRP registrations are propagated from other ports. Setting this to a value of enabled(1) will be stored by the agent but will only take effect on the GMRP protocol operation if dot1dGmrpStatus also indicates the value enabled(1). This object affects all GMRP Applicant and Registrar state machines on this port. A transition from disabled(2) to enabled(1) will cause a reset of all GMRP state machines on this port.

DEFVAL-enabled

5.16.4.4.1.2 dot1dPortGmrpFailedRegistrations

The total number of failed GMRP registrations, for any reason, in all VLANs, on this port.

5.16.4.4.1.3 dot1dPortGmrpLastPduOrigin

The Source MAC Address of the last GMRP message received on this port.

5.16.4.5 dot1dGmrp Group

This group contains the objects for configuring and reporting on operation of the GARP Multicast Registration Protocol (GMRP).

Valid Object Value(s)

This group is not supported in the BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

5.16.4.6 dot1dTp Group

This group contains objects that describe the entity's state with respect to transparent bridging.

OID:

- dot1dTp - { dot1dBridge 4 } 1.3.6.1.2.1.17.4

5.16.4.6.1 dot1dTpHCPortTable

The High Capacity Port Table extends the dot1dTp group form the original bridge MIB and contains the objects for reporting port bridging statistics for high capacity network interfaces. This table contains information about every high capacity port that is associated with this transparent bridge.

Index:

- dot1dTpPort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1dTpHCPortTable - { dot1dTp 5 } 1.3.6.1.2.1.17.4.5
- dot1dTpHCPortEntry - { dot1dTpHCPortTable 1 } 1.3.6.1.2.1.17.4.5.1

Table 161. dot1dTpHCPortTable

Object	OID	Syntax	Access	Status	Supported
dot1dTpHCPortInFrames	{ dot1dTpHCPortEntry 1 }	Counter64	read-only	current	N
dot1dTpHCPortOutFrames	{ dot1dTpHCPortEntry 2 }	Counter64	read-only	current	N
dot1dTpHCPortInDiscards	{ dot1dTpHCPortEntry 3 }	Counter64	read-only	current	N

5.16.4.6.1.1 dot1dTpHCPortInFrames

The number of frames that have been received by this port from its segment. Note that a frame received on the interface corresponding to this port is only counted by this object if and only if it is for a protocol being processed by the local bridging function, including bridge management frames.

5.16.4.6.1.2 dot1dTpHCPortOutFrames

The number of frames that have been transmitted by this port to its segment. Note that a frame transmitted on the interface corresponding to this port is only counted by this object if and only if it is for a protocol being processed by the local bridging function, including bridge management frames.

5.16.4.6.1.3 dot1dTpHCPortInDiscards

Count of valid frames that have been received by this port from its segment which were discarded (that is, filtered) by the Forwarding Process.

5.16.4.6.2 dot1dTpPortOverflowTable

The Upper part of High Capacity Port Table for Transparent Bridges. This is a table that contains the most-significant bits of statistics counters for ports that are associated with this transparent bridge that are on high capacity interfaces, as defined in the conformance clauses for this table. This table is provided as a way to read 64-bit counters for agents which support only SNMPv1.

Note that the reporting of most-significant and least-significant counter bits separately runs the risk of missing an overflow of the lower bits in the interval between sampling. The manager must be aware of this possibility, even within the same varbindlist, when interpreting the results of a request or asynchronous notification.

This table is a sequence of dot1dTpPortOverflowEntry which consists of the most significant bits of statistics counters for a high capacity interface of a transparent bridge. Each object is associated with a corresponding object in dot1dTpPortTable which indicates the least significant bits of the counter.

Index:

- dot1dTpPort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1dTpPortOverflowTable - { dot1dTp 6 } 1.3.6.1.2.1.17.4.6
- dot1dTpPortOverflowEntry - { dot1dTpPortOverflowTable 1 } 1.3.6.1.2.1.17.4.6.1

Table 162. dot1dTpPortOverflowTable

Object	OID	Syntax	Access	Status	Supported
dot1dTpTpPortInOverflowFrames	{ dot1dTpPortOverflowEntry 1 }	Counter32	read-only	current	N
dot1dTpTpPortOutOverflowFrames	{ dot1dTpPortOverflowEntry 2 }	Counter32	read-only	current	N
dot1dTpTpPortInOverflowDiscards	{ dot1dTpPortOverflowEntry 3 }	Counter32	read-only	current	N

5.16.4.6.2.1 dot1dTpPortInOverflowFrames

The number of times the associated dot1dTpPortInFrames (Section 5.15.4.4.4.3, Page 263) counter has overflowed.

5.16.4.6.2.2 dot1dTpPortOutOverflowFrames

The number of times the associated dot1dTpPortOutFrames (Section 5.15.4.4.4.4, Page 263) counter has overflowed.

5.16.4.6.2.3 dot1dTpPortInOverflowDiscards

The number of times the associated dot1dTpPortInDiscards (Section 5.15.4.4.4.5, Page 263) counter has overflowed.

5.16.5 Conformance Information

5.16.5.1 Compliance

5.16.5.1.1 pBridgeCompliance

The compliance statement for device support of Priority and Multicast Filtering extended bridging services.

Mandatory Groups:

- pBridgeExtCapGroup

Optional Groups:

- pBridgeDeviceGmrpGroup - This group is mandatory for devices supporting the GMRP application, defined by IEEE 802.1D Extended Filtering Services.
- pBridgeDevicePriorityGroup - This group is mandatory only for devices supporting the priority forwarding operations defined by IEEE 802.1D.
- pBridgeDefaultPriorityGroup - This group is mandatory only for devices supporting the priority forwarding operations defined by the extended bridge services with media types, such as Ethernet, that do not support native User Priority.
- pBridgeRegenPriorityGroup - This group is mandatory only for devices supporting the priority forwarding operations defined by IEEE 802.1D and which have interface media types that support native User Priority, such as IEEE 802.5.
- pBridgePriorityGroup - This group is mandatory only for devices supporting the priority forwarding operations defined by IEEE 802.1D.
- pBridgeAccessPriorityGroup - This group is optional and is relevant only for devices supporting the priority forwarding operations defined by IEEE 802.1D and which have interface media types that support native Access Priority, such as IEEE 802.5.
- pBridgePortGarpGroup - This group is mandatory for devices supporting any of the GARP applications, including GMRP, defined by the extended filtering services of 802.1D, and GVRP, defined by 802.1Q (refer to the Q-BRIDGE-MIB for conformance statements for GVRP).
- pBridgePortGmrpGroup - This group is mandatory for devices supporting the GMRP application, as defined by IEEE 802.1D Extended Filtering Services.
- pBridgeHCPortGroup - Support for this group in a device is mandatory for those bridge ports which map to network interfaces that have the value of the corresponding instance of ifSpeed greater than 650,000,000 bits/second.
- pBridgePortOverflowGroup - Support for this group in a device is mandatory for those bridge ports which map to network interfaces that have the value of the corresponding instance of ifSpeed greater than 650,000,000 bits/second.

Objects that have a MIN-ACCESS defined:

- dot1dPortNumTrafficClasses - MIN-ACCESS read-only
  Write access is not required.

- dot1dTrafficClass - MIN-ACCESS read-only
  Write access is not required.
- dot1dRegenUserPriority - MIN-ACCESS read-only
  Write access is not required.

5.16.5.2 Conformance

Table 163. P-BRIDGE-MIB Conformance (1 of 2)

Group	Description	Objects	Supported
pBridgeExtCapGroup	A collection of objects indicating the optional capabilities of the device.	dot1dDeviceCapabilities	Y
		dot1dPortCapabilities	Y
pBridgeDeviceGmrpGroup	A collection of objects providing device-level control for the Multicast Filtering extended bridge services.	dot1dGmrpStatus	Y
pBridgeDevicePriorityGroup	A collection of objects providing device-level control for the Priority services.	dot1dTrafficClassesEnabled	Y
pBridgeDefaultPriorityGroup	A collection of objects defining the UserPriority applicable to each port for media which do not support native user priority.	dot1dPortDefaultUserPriority	Y
pBridgeRegenPriorityGroup	A collection of objects defining User Priorities applicable to each port for media which support native User Priority.	dot1dRegenUserPriority	N
pBridgePriorityGroup	A collection of objects defining the traffic classes within a bridge for each evaluated User Priority.	dot1dPortNumTrafficClasses	Y
		dot1dTrafficClass	Y
pBridgeAccessPriorityGroup	A collection of objects defining the media dependent outbound access level for each priority.	dot1dPortOutboundAccessPriority	N
pBridgePortGarpGroup	A collection of objects providing port level control and status information for GARP operation.	dot1dPortGarpJoinTime	N
		dot1dPortGarpLeaveTime	N
		dot1dPortGarpLeaveAllTime	N
pBridgePortGmrpGroup	A collection of objects providing port level control and status information for GMRP operation.	dot1dPortGmrpStatus	N
		dot1dPortGmrpFailedRegistrations	N
		dot1dPortGmrpLastPduOrigin	N

Table 163. P-BRIDGE-MIB Conformance (2 of 2)

Group	Description	Objects	Supported
pBridgeHCPortGroup	A collection of objects providing 64-bit statistics counters for high capacity bridge ports.	dot1dTpHCPortInFrames	N
		dot1dTpHCPortOutFrames	N
		dot1dTpHCPortInDiscards	N
pBridgePortOverflowGroup	A collection of objects providing overflow statistics counters for high capacity bridge ports.	dot1dTpPortInOverflowFrames	N
		dot1dTpPortOutOverflowFrames	N
		dot1dTpPortInOverflowDiscards	N

5.17 Q-BRIDGE-MIB — RFC 2674

5.17.1 Introduction

The VLAN Bridge MIB module for managing Virtual Bridged Local Area Networks, as defined by IEEE 802.1Q-1998. The LAST-UPDATED for this MIB in the RFC was “9908250000Z.”

5.17.2 Textual Conventions

- PortList - Each octet within this value specifies a set of eight ports, with the first octet specifying ports 1 through 8, the second octet specifying ports 9 through 16, etc. Within each octet, the most significant bit represents the lowest numbered port, and the least significant bit represents the highest numbered port. Thus, each port of the bridge is represented by a single bit within the value of this object. If that bit has a value of “1” then that port is included in the set of ports; the port is not included if its bit has a value of “0.” Its syntax is an OCTET STRING.
See Implementation Notes on PortList (Section 5.17.2.1.1, Page 279)

- VlanIndex - A value used to index per-VLAN tables: values of 0 and 4095 are not permitted; if the value is between 1 and 4094 inclusive, it represents an IEEE 802.1Q VLAN-ID with global scope within a given bridged domain (see VlanId textual convention). If the value is greater than 4095 then it represents a VLAN with scope local to the particular agent, that is one without a global VLAN-ID assigned to it. Such VLANs are outside the scope of IEEE 802.1Q but it is convenient to be able to manage them in the same way using this MIB. Its syntax is an Unsigned32.

- VlanId - A 12-bit VLAN ID used in the VLAN Tag header. Its syntax is an INTEGER (1…4094).

5.17.2.1 Implementation Notes

5.17.2.1.1 PortList

Table 164. Implementation for PortList

Octet Nbr.	Port	Bit Position							
		1 (MSB)	2	3	4	5	6	7	8 (LSB)
1	DSL Port 1	PIWG 1	PIWG 2	PIWG 3	PIWG 4	PIWG 5	PIWG 6	PIWG 7	PIWG 8
2	DSL Port 2	PIWG 1	PIWG 2	PIWG 3	PIWG 4	PIWG 5	PIWG 6	PIWG 7	PIWG 8
⋮	⋮	…	…	…	…	…	…	…	…
48	DSL Port 48	PIWG 1	PIWG 2	PIWG 3	PIWG 4	PIWG 5	PIWG 6	PIWG 7	PIWG 8

Table 164. Implementation for PortList

Octet Nbr.	Port	Bit Position							
		1 (MSB)	2	3	4	5	6	7	8 (LSB)
49	Management Ethernet Interfaces	Mgmt IB Eth	0	0	0	0	0	0	0
50	Ethernet Port 1…3	0	Eth 2	Eth 3	0	0	0	0	0

5.17.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- dot1dBridge - { mib-2 17 } 1.3.6.1.2.1.17
- qBridgeMIB - { dot1dBridge 7 } 1.3.6.1.2.1.17.7
- qBridgeMIBObjects - { qBridgeMIB 1 } 1.3.6.1.2.1.17.7.1
- dot1qBase - { qBridgeMIBObjects 1 } 1.3.6.1.2.1.17.7.1.1
- dot1qTp - { qBridgeMIBObjects 2 } 1.3.6.1.2.1.17.7.1.2
- dot1qStatic - { qBridgeMIBObjects 3 } 1.3.6.1.2.1.17.7.1.3
- dot1qVlan - { qBridgeMIBObjects 4 } 1.3.6.1.2.1.17.7.1.4

5.17.4 MIB Objects

Table 165. Q-BRIDGE-MIB Objects

Object	OID	Description	Supported
dot1qBase	{ qBridgeMIBObjects 1 }	This group contains objects which are applicable to all bridges implementing IEEE 802.1Q virtual LANs.	Y

Table 165. Q-BRIDGE-MIB Objects

Object	OID	Description	Supported
dot1qTp	{ qBridgeMIBObjects 2 }	This group contains objects that control the operation and report the status of transparent bridging. This includes management of the Dynamic-Filtering Databases for both unicast and multicast forwarding. This group also defines a dot1dTpFdbTable (Section 5.15.4.4.3, Page 261) that allowed multiple forwarding databases for 802.1Q transparent devices. This version of the table is an alternative to the one defined for 802.1D devices which only supports a single forwarding database. Refer to the Bridge MIB for the single forwarding database version of the table.	Y
dot1qStatic	{ qBridgeMIBObjects 3 }	This group defines the Static Destination-Address Filtering Database and contains objects that control static configuration information for transparent bridging. This includes management of the static entries in the Filtering Databases for both unicast and multicast forwarding.	Y
dot1qVlan	{ qBridgeMIBObjects 4 }	This group contains objects that control configuration and report status of the Virtual LANs known to a bridge. This includes management of the statically configured VLANs as well as reporting VLANs discovered by other means, such as GVRP. It also controls configuration and reports status of per-port objects relating to VLANs and reports traffic statistics. It also provides for management of the VLAN Learning Constraints.	Y

5.17.4.1 dot1qBase Group

This group contains objects which are applicable to all bridges implementing IEEE 802.1Q virtual LANs.

OIDs:

- dot1qBase - { qBridgeMIBObjects 1 } 1.3.6.1.2.1.17.7.1.1

Table 166. dot1qBase

Object	OID	Syntax	Access	Status	Supported
dot1qVlanVersionNumber	{ dot1qBase 1 }	INTEGER version1(1)	read-only	current	Y
dot1qMaxVlanId	{ dot1qBase 2 }	VlanId	read-only	current	Y
dot1qMaxSupportedVlans	{ dot1qBase 3 }	Unsigned32	read-only	current	Y
dot1qNumVlans	{ dot1qBase 4 }	Unsigned32	read-only	current	Y
dot1qGvrpStatus	{ dot1qBase 5 }	EnabledStatus	read-only[a]	current	Y

a. Product Imposed Limit

5.17.4.1.1 dot1qVlanVersionNumber

The version number of IEEE 802.1Q that this device supports.

Valid Object Value(s)

- version1(1)

5.17.4.1.2 dot1qMaxVlanId

The maximum IEEE 802.1Q VLAN ID that this device supports.

Valid Object Value(s)

- 4095

5.17.4.1.3 dot1qMaxSupportedVlans

The maximum number of IEEE 802.1Q VLANs that this device supports.

- The following formula determines the value returned

$$(NbrDslPortsPerChassis \times NbrPiwgsPerDslPort \times NbrVlansPerPiwg) + InBandMgmtVlan$$

Where:

NbrDslPortsPerChassis		=	24 or 48
NbrPiwgsPerDslPort		=	8
NbrVlansPerPiwg		=	8
Final Result:	24 port chassis	=	(24 x 8 x 8) + 1 = 1537
	48 port chassis	=	(48 x 8 x 8) + 1 = 3073

5.17.4.1.4 dot1qNumVlans

The current number of IEEE 802.1Q VLANs that are configured in this device.

5.17.4.1.5 dot1qGvrpStatus

The administrative status requested by management for GVRP. The value enabled(1) indicates that GVRP should be enabled on this device, on all ports for which it has not been specifically disabled. When disabled(2), GVRP is disabled on all ports and all GVRP packets are forwarded transparently. This object affects all GVRP Applicant and Registrar state machines. A transition from disabled(2) to enabled(1) will cause a reset of all GVRP state machines on all ports.

Valid Object Value(s)

- disabled(2)

5.17.4.2 dot1qTp Group

This group contains objects that control the operation and report the status of transparent bridging. This includes management of the dynamic Filtering Databases for both unicast and multicast forwarding.

OIDs:

- dot1qTp - { qBridgeMIBObjects 2 } 1.3.6.1.2.1.17.7.1.2

Table 167. dot1qTp

Object	OID	Syntax	Access	Status	Supported
dot1qFdbTable	{ dot1qTp 1 }	SEQUENCE of dot1qFdbEntry	not-accessible	current	N
dot1qTpFdbTable	{ dot1qTp 2 }	SEQUENCE of dot1qTpFdbEntry	not-accessible	current	N

Table 167. dot1qTp

Object	OID	Syntax	Access	Status	Supported
dot1qTpGroupTable	{ dot1qTp 3 }	SEQUENCE of dot1qTpGroupEntry	not-accessible	current	N
dot1qForwardAllTable	{ dot1qTp 4 }	SEQUENCE of dot1qForwardAllEntry	not-accessible	current	N
dot1qForwardUnregisteredTable	{ dot1qTp 5 }	SEQUENCE of dot1qForwardUnregisteredEntry	not-accessible	current	N

5.17.4.2.1 dot1qFdbTable

The Filtering Database Table contains configuration and control information for each Filtering Database currently operating on this device. Entries in this table appear automatically when VLANs are assigned FDB IDs in the dot1qVlanCurrentTable (Section 5.17.4.4.2, Page 291).

Index:

- dot1qFdbId

OIDs:

- dot1qFdbTable - { dot1qTp 1 } 1.3.6.1.2.1.17.7.1.2.1
- dot1qFdbEntry - { dot1qFdbTable 1 } 1.3.6.1.2.1.17.7.1.2.1.1

Table 168. dot1qFdbTable

Object	OID	Syntax	Access	Status	Supported
dot1dFdbId	{ dot1qFdbEntry 1 }	Unsigned32	not-accessible	current	N
dot1dTpFdbDynamicCount	{ dot1qFdbEntry 2 }	Counter32	read-only	current	N

5.17.4.2.1.1 dot1qFdbId

The identity of this Filtering Database.

5.17.4.2.1.2 dot1qFdbDynamicCount

The current number of dynamic entries in this Filtering Database.

5.17.4.2.2 dot1qTpFdbTable

Transparent Bridge Filtering Database Table contains information about unicast entries for which the device has forwarding and/or filtering information. This information is used by the transparent bridging function in determining how to propagate a received frame.

Each entry consists of information about a specific unicast MAC address for which the device has some forwarding and/or filtering information.

Note in RFC:

Multiple Forwarding Databases for 802.1Q Transparent devices

This table is an alternative to the dot1dTpFdbTable (Section 5.15.4.4.3, Page 261) previously defined for 802.1D devices which only support a single Forwarding Database.

Indices:

- dot1qFdbId (Section 5.17.4.2.1.1, Page 283)
- dot1qTpFdbAddress

OIDs:

- dot1dTpFdbTable - { dot1qTp 2 } 1.3.6.1.2.1.17.7.1.2.2
- dot1qTpFdbEntry - { dot1dTpFdbTable 1 } 1.3.6.1.2.1.17.7.1.2.2.1

Table 169. dot1qTpFdbTable

Object	OID	Syntax	Access	Status	Supported
dot1qTpFdbAddress	{ dot1qTpFdbEntry 1 }	MacAddress	not-accessible	current	Y
dot1qTpFdbPort	{ dot1qTpFdbEntry 2 }	INTEGER (0…65535)	read-only	current	Y
dot1qTpFdbStatus	{ dot1qTpFdbEntry 3 }	INTEGER other (1) invalid (2) learned (3) self (4) mgmt (5)	read-only	current	Y

5.17.4.2.2.1 dot1qTpFdbAddress

A unicast MAC address for which the device has forwarding and/or filtering information.

5.17.4.2.2.2 dot1qTpFdbPort

Either the value “0,” or the port number of the port on which a frame having a source address equal to the value of the corresponding instance of dot1qTpFdbAddress has been seen. A value of “0” indicates that the port number has not been learned but that the device does have some forwarding/filtering information about this address (for example, in the dot1qStaticUnicastTable [dot1qStaticUnicastTable, Section 5.17.4.3.1, Page 287].

Implementors are encouraged to assign the port value to this object whenever it is learned even for addresses for which the corresponding value of dot1qTpFdbStatus is not learned(3).

5.17.4.2.2.3 dot1qTpFdbStatus

The status of this entry. The meanings of the values are:

- other(1) - none of the following. This may include the case where some other MIB object (not the corresponding instance of dot1qTpFdbPort, nor an entry in the dot1qStaticUnicastTable) is being used to determine if and how frames addressed to the value of the corresponding instance of dot1qTpFdbAddress are being forwarded.
- invalid(2) - this entry is no longer valid (for example, it was learned but has since aged out), but has not yet been flushed from the table.
- learned(3) - the value of the corresponding instance of dot1qTpFdbPort was learned and is being used.
- self(4) - the value of the corresponding instance of dot1qTpFdbAddress represents one of the device's addresses. The corresponding instance of dot1qTpFdbPort indicates which of the device's ports has this address.
- mgmt(5) - the value of the corresponding instance of dot1qTpFdbAddress is also the value of an existing instance of dot1qStaticAddress (*Author comment: I think that this is a typo in the RFC. I seems like this should be dot1qStaticUnicastAddress [Section 5.17.4.3.1.1, Page 288] as there is currently no object called dot1qStaticAddress.)*

5.17.4.2.3 dot1qTpGroupTable

This is the Dynamic Group Registration Table which contains filtering information for VLANs configured into the bridge by (local or network) management, or learnt dynamically, specifying the set of ports to which frames received on a VLAN for this FDB and containing a specific Group destination address are allowed to be forwarded.

Each entry consists of filtering information configured into the bridge by management, or learnt dynamically, specifying the set of ports to which frames received on a VLAN and containing a specific Group destination address, are allowed to be forwarded. The subset of these ports learnt dynamically is also provided.

Indices:

- dot1qVlanIndex (Reference Section 5.17.4.4.2.2, Page 291)
- dot1qTpGroupAddress

OIDs:

- dot1dTpGroupTable - { dot1qTp 3 } 1.3.6.1.2.1.17.7.1.2.3
- dot1qTpGroupEntry - { dot1dTpGroupTable 1 } 1.3.6.1.2.1.17.7.1.2.3.1

Table 170. dot1qTpGroupTable

Object	OID	Syntax	Access	Status	Supported
dot1qTpGroupAddress	{ dot1qTpGroupEntry 1 }	MacAddress	not-accessible	current	N
dot1qTpGroupEgressPorts	{ dot1qTpGroupEntry 2 }	PortList	read-only	current	N
dot1qTpGroupLearnt	{ dot1qTpGroupEntry 3 }	PortList	read-only	current	N

5.17.4.2.3.1 dot1qTpGroupAddress

The destination Group MAC address in a frame to which this entry's filtering information applies.

5.17.4.2.3.2 dot1qTpGroupEgressPorts

The complete set of ports, in this VLAN, to which frames destined for this Group MAC address are currently being explicitly forwarded. This does not include ports for which this address is only implicitly forwarded, in the dot1qForwardAllPorts list.

5.17.4.2.3.3 dot1qTpGroupLearnt

The subset of ports in dot1qTpGroupEgressPorts which were learnt by GMRP **or some other dynamic mechanism**, in this Filtering database.

5.17.4.2.4 dot1qForwardAllTable

This is a table containing forwarding information for each VLAN, specifying the set of ports to which forwarding of all multicasts applies, configured statically by management or dynamically by GMRP. An entry appears in this table for all VLANs that are currently instantiated.

Each entry consists of forwarding information for a VLAN, specifying the set of ports to which all multicasts should be forwarded, configured statically by management or dynamically by GMRP.

Index:

- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)

OIDs:

- dot1qForwardAllTable - { dot1qTp 4 } 1.3.6.1.2.1.17.7.1.2.4
- dot1qForwardAllEntry - { dot1qForwardAllTable 1 } 1.3.6.1.2.1.17.7.1.2.4.1

Table 171. dot1qForwardAllTable

Object	OID	Syntax	Access	Status	Supported
dot1qForwardAllPorts	{ dot1qForwardAllEntry 1 }	PortList	not-accessible	current	N
dot1qForwardAllStaticPorts	{ dot1qForwardAllEntry 2 }	PortList	read-write	current	N
dot1qForwardAllForbiddenPorts	{ dot1qForwardAllEntry 3 }	PortList	read-write	current	N

5.17.4.2.4.1 dot1qForwardAllPorts

The complete set of ports in this VLAN to which all multicast group-addressed frames are to be forwarded. This includes ports for which this need has been determined dynamically by GMRP, or configured statically by management.

5.17.4.2.4.2 dot1qForwardAllStaticPorts

The set of ports configured by management in this VLAN to which all multicast group-addressed frames are to be forwarded. Ports entered in this list will also appear in the complete set shown by dot1qForwardAllPorts. This value is restored after the device is reset. This only applies to ports that are members of the VLAN, defined by dot1qVlanCurrentEgressPorts. A port may not be added in this set if it is already a member of the set of ports in dot1qForwardAllForbiddenPorts. The default value is a string of ones of appropriate length, to indicate standard non-EFS behavior, that is, forward all multicasts to all ports.

5.17.4.2.4.3 dot1qForwardAllForbiddenPorts

The set of ports configured by management in this VLAN for which the Service Requirement attribute Forward All Multicast Groups may not be dynamically registered by GMRP. This value is restored after the device is reset. A port may not be added in this set if it is already a member of the set of ports in dot1qForwardAllStaticPorts. The default value is a string of zeros of appropriate length.

5.17.4.2.5 dot1qForwardUnregisteredTable

This is a table containing forwarding information for each VLAN, specifying the set of ports to which forwarding of multicast group-addressed frames for which there is no more specific forwarding information applies. This is configured statically by management and determined dynamically by GMRP. An entry appears in this table for all VLANs that are currently instantiated.

Each entry consists of forwarding information for a VLAN, specifying the set of ports to which all multicasts for which there is no more specific forwarding information shall be forwarded. This is configured statically by management or dynamically by GMRP.

Index:

- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)

OIDs:

- dot1qForwardUnregisteredTable - { dot1qTp 5 } 1.3.6.1.2.1.17.7.1.2.5
- dot1qForwardUnregisteredEntry - { dot1qForwardUnregisteredTable 1 } 1.3.6.1.2.1.17.7.1.2.5.1

Table 172. dot1qForwardUnregisteredTable

Object	OID	Syntax	Access	Status	Supported
dot1qForwardUnregisteredPorts	{ dot1qForwardUnregisteredEntry 1 }	PortList	read-only	current	N

Table 172. dot1qForwardUnregisteredTable

Object	OID	Syntax	Access	Status	Supported
dot1qForwardUnregisteredStaticPorts	{ dot1qForwardUnregisteredEntry 2 }	PortList	read-write	current	N
dot1qForwardUnregisteredForbiddenPorts	{ dot1qForwardUnregisteredEntry 3 }	PortList	read-write	current	N

5.17.4.2.5.1 dot1qForwardUnregisteredPorts

The complete set of ports in this VLAN to which multicast group-addressed frames for which there is no more specific forwarding information is forwarded. This includes ports for which this need has been determined dynamically by GMRP, or configured statically by management.

5.17.4.2.5.2 dot1qForwardUnregisteredStaticPorts

The set of ports configured by management, in this VLAN, to which multicast group-addressed frames for which there is no more specific forwarding information are to be forwarded. Ports entered in this list will also appear in the complete set shown by dot1qForwardUnregisteredPorts. This value is restored after the device is reset. A port may not be added in this set if it is already a member of the set of ports in dot1qForwardUnregisteredForbiddenPorts. The default value is a string of zeros of appropriate length, although this has no effect with the default value of dot1qForwardAllStaticPorts (Section 5.17.4.2.4.2, Page 286).

5.17.4.2.5.3 dot1qForwardUnregisteredForbiddenPorts

The set of ports configured by management in this VLAN for which the Service Requirement attribute Forward Unregistered Multicast Groups may not be dynamically registered by GMRP. This value is restored after the device is reset. A port may not be added in this set if it is already a member of the set of ports in dot1qForwardUnregisteredStaticPorts. The default value is a string of zeros of appropriate length.

5.17.4.3 dot1qStatic Group

This group contains objects that control static configuration information for transparent bridging. This includes management of the static entries in the Filtering Databases for both unicast and multicast forwarding.

OIDs:

- dot1qStatic - { qBridgeMIBObjects 3 } 1.3.6.1.2.1.17.7.1.3

Table 173. dot1qStatic

Object	OID	Syntax	Access	Status	Supported
dot1qStaticUnicastTable	{ dot1qStatic 1 }	SEQUENCE of dot1qStaticUnicastEntry	not-accessible	current	Y
dot1qStaticMulticastTable	{ dot1qStatic 2 }	SEQUENCE of dot1qStaticMulticastEntry	not-accessible	current	N

5.17.4.3.1 dot1qStaticUnicastTable

The Static (Destination-Address Filtering) Database table containing filtering information for Unicast MAC addresses for each Filtering Database, configured into the device by (local or network) management specifying the set of ports to which frames received from specific ports and containing specific unicast destination addresses are allowed to be forwarded. A value of zero in this table as the port number from which frames with a specific destination address are received, is used to specify all ports for which there is no specific entry in this table for that particular destination address. Entries are valid for unicast addresses only.

Each entry consists of filtering information configured into the device by (local or network) management specifying the set of ports to which frames received from a specific port and containing a specific unicast destination address are allowed to be forwarded.

Indices:

- dot1qFdbId (Section 5.17.4.2.1.1, Page 283)
- dot1qStaticUnicastAddress
- dot1qStaticUnicastReceivePort

OIDs:

- dot1qStaticUnicastTable - { dot1qStatic 1 } 1.3.6.1.2.1.17.7.1.3.1
- dot1qStaticUnicastEntry - { dot1qStaticUnicastTable 1 } 1.3.6.1.2.1.17.7.1.3.1.1

Table 174. dot1qStaticUnicastTable

Object	OID	Syntax	Access	Status	Supported
dot1qStaticUnicastAddress	{ dot1qStaticUnicastEntry 1 }	MacAddress	not-accessible	current	Y
dot1qStaticUnicastReceivePort	{ dot1qStaticUnicastEntry 2 }	INTEGER (0…65535)	not-accessible	current	Y
dot1qStaticUnicastAllowedToGoTo	{ dot1qStaticUnicastEntry 3 }	PortList	read-write	current	Y
dot1qStaticUnicastStatus	{ dot1qStaticUnicastEntry 4 }	INTEGER other (1) invalid (2) permanent (3) deleteOnReset (4) deleteOnTimeout (5)	read-write	current	Y

5.17.4.3.1.1 dot1qStaticUnicastAddress

The destination MAC address in a frame to which this entry's filtering information applies. This object must take the value of a unicast address.

5.17.4.3.1.2 dot1qStaticUnicastReceivePort

Either the value “0,” or the port number of the port from which a frame must be received in order for this entry's filtering information to apply. A value of zero indicates that this entry applies on all ports of the device for which there is no other applicable entry.

5.17.4.3.1.3 dot1qStaticUnicastAllowedToGoTo

The set of ports for which a frame with a specific unicast address will be flooded in the event that it has not been learned. It also specifies the set of ports a specific unicast address may be dynamically learnt on. The dot1qFdbTable (Section 5.17.4.2.1, Page 283) will have an equivalent entry with a dot1qTpFdbPort (Section 5.17.4.2.2.2, Page 284) value of “0” until this address has been learned, when it will be updated with the port the address has been seen on. This only applies to ports that are members of the VLAN, defined by dot1qVlanCurrentEgressPorts (Section 5.17.4.4.2.4, Page 292). The default value of this object is a string of ones of appropriate length.

5.17.4.3.1.4 dot1qStaticUnicastStatus

This object indicates the status of this entry. The default value is permanent.

- other(1) - this entry is currently in use but the conditions under which it will remain so differ from the following values.
- invalid(2) - writing this value to the object removes the corresponding entry.
- permanent(3) - this entry is currently in use and will remain so after the next reset of the bridge.
- deleteOnReset(4) - this entry is currently in use and will remain so until the next reset of the bridge.

- deleteOnTimeout(5) - this entry is currently in use and will remain so until it is aged out.

5.17.4.3.2 dot1qStaticMulticastTable

This is a table containing filtering information for Multicast and Broadcast MAC addresses for each VLAN, configured into the device by (local or network) management specifying the set of ports to which frames received from specific ports and containing specific Multicast and Broadcast destination addresses are allowed to be forwarded. A value of zero in this table as the port number from which frames with a specific destination address are received, is used to specify all ports for which there is no specific entry in this table for that particular destination address. Entries are valid for Multicast and Broadcast addresses only.

Each entry consists of filtering information configured into the device by (local or network) management specifying the set of ports to which frames received from this specific port for this VLAN and containing this Multicast or Broadcast destination address are allowed to be forwarded.

Indices:

- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)
- dot1qStaticMulticastAddress
- dot1qStaticMulticastReceivePort

OIDs:

- dot1qStaticMulticastTable - { dot1qStatic 2 } 1.3.6.1.2.1.17.7.1.3.2
- dot1qStaticMulticastEntry - { dot1qStaticMulticastTable 1 } 1.3.6.1.2.1.17.7.1.3.2.1

Table 175. dot1qStaticMulticastTable

Object	OID	Syntax	Access	Status	Supported
dot1qStaticMulticastAddress	{ dot1qStaticMulticastEntry 1 }	MacAddress	not-accessible	current	N
dot1qStaticMulticastReceivePort	{ dot1qStaticMulticastEntry 2 }	INTEGER (0…65535)	not-accessible	current	N
dot1qStaticMulticastStaticEgressPorts	{ dot1qStaticMulticastEntry 3 }	PortList	read-write	current	N
dot1qStaticMulticastForbiddenEgressPorts	{ dot1qStaticMulticastEntry 4 }	PortList	read-write	current	N
dot1qStaticMulticastStatus	{ dot1qStaticMulticastEntry 5 }	INTEGER other (1) invalid (2) permanent (3) deleteOnReset (4) deleteOnTimeout (5)	read-write	current	N

5.17.4.3.2.1 dot1qStaticMulticastAddress

The destination MAC address in a frame to which this entry's filtering information applies. This object must take the value of a Multicast or Broadcast address.

5.17.4.3.2.2 dot1qStaticMulticastReceivePort

Either the value “0,” or the port number of the port from which a frame must be received in order for this entry's filtering information to apply. A value of zero indicates that this entry applies on all ports of the device for which there is no other applicable entry.

5.17.4.3.2.3 dot1qStaticMulticastStaticEgressPorts

The set of ports to which frames received from a specific port and destined for a specific Multicast or Broadcast MAC address must be forwarded, regardless of any dynamic information such as that from GMRP. A port may not be added in this set if it is already a member of the set of ports in dot1qStaticMulticastForbiddenEgressPorts. The default value of this object is a string of ones of appropriate length.

5.17.4.3.2.4 dot1qStaticMulticastForbiddenEgressPorts

The set of ports to which frames received from a specific port and destined for a specific Multicast or Broadcast MAC address must not be forwarded, regardless of any dynamic information such as that from GMRP. A port may not be added in this set if it is already a member of the set of ports in dot1qStaticMulticastStaticEgressPorts.

The default value of this object is a string of zeros of appropriate length.

5.17.4.3.2.5 dot1qStaticMulticastStatus

This object indicates the status of this entry. Its default value is permanent.

- other(1) - this entry is currently in use but the conditions under which it will remain so differ from the following values.
- invalid(2) - writing this value to the object removes the corresponding entry.
- permanent(3) - this entry is currently in use and will remain so after the next reset of the bridge.
- deleteOnReset(4) - this entry is currently in use and will remain so until the next reset of the bridge.
- deleteOnTimeout(5) - this entry is currently in use and will remain so until it is aged out.

5.17.4.4 dot1qVlan Group

This group contains objects that control configuration and report status of the Virtual LANs known to a bridge. This includes management of the statically configured VLANs as well as reporting VLANs discovered by other means, such as GVRP. It also controls configuration and reports status of per-port objects relating to VLANs and reports traffic statistics. It also provides for management of the VLAN Learning Constraints.

OIDs:

- dot1qVlan - { qBridgeMIBObjects 4 } 1.3.6.1.2.1.17.7.1.4

Table 176. dot1qVlan

Object	OID	Syntax	Access	Status	Supported
dot1qVlanNumDeletes	{ dot1qVlan 1 }	Counter32	Read-only	current	N
dot1qVlanCurrentTable	{ dot1qVlan 2 }	SEQUENCE of dot1qVlanCurrentEntry	not-accessible	current	Y
dot1qVlanStaticTable	{ dot1qVlan 3 }	SEQUENCE of dot1qVlanStaticEntry	not-accessible	current	Y
dot1qNextFreeLocalVlanIndex	{ dot1qVlan 4 }	INTEGER(0 | 4096…2147483647)	read-only	current	Y
dot1qPortVlanTable	{ dot1qVlan 5 }	SEQUENCE of dot1qPortVlanEntry	not-accessible	current	N
dot1qPortVlanStatisticsTable	{ dot1qVlan 6 }	SEQUENCE of dot1qPortVlanStatisticsEntry	not-accessible	current	N
dot1qPortVlanHCStatisticsTable	{ dot1qVlan 7 }	SEQUENCE of dot1qPortVlanHCStatisticsEntry	not-accessible	current	N
dot1qLearningConstraintsTable	{ dot1qVlan 8 }	SEQUENCE of dot1qLearningConstraintsEntry	not-accessible	current	N
dot1qConstraintSetDefault	{ dot1qVlan 9 }	INTEGER (0…65535)	read-write	current	N
dot1qConstraintTypeDefault	{ dot1qVlan 10 }	INTEGER independent (1) shared (2)	read-write	current	N

5.17.4.4.1 dot1qVlanNumDeletes

The number of times a VLAN entry has been deleted from the dot1qVlanCurrentTable (Section 5.17.4.4.2, Page 291) (for any reason). If an entry is deleted, then inserted, and then deleted, this counter is incremented by 2.

5.17.4.4.2 dot1qVlanCurrentTable

The Current VLAN Database table contains current configuration information for each VLAN currently configured into the device by (local or network) management, or dynamically created as a result of GVRP requests received.

Each entry consists of information for a VLAN configured into the device by (local or network) management, or dynamically created as a result of GVRP requests received.

Indices:

- dot1qVlanTimeMark
- dot1qVlanIndex

OIDs:

- dot1qVlanCurrentTable - { dot1qVlan 2 } 1.3.6.1.2.1.17.7.1.4.2
- dot1qVlanCurrentEntry - { dot1qVlanCurrentTable 1 } 1.3.6.1.2.1.17.7.1.4.2.1

Table 177. dot1qVlanCurrentTable

Object	OID	Syntax	Access	Status	Supported
dot1qVlanTimeMark	{ dot1qVlanCurrentEntry 1 }	TimeFilter	not-accessible	current	Y
dot1qVlanIndex	{ dot1qVlanCurrentEntry 2 }	VlanIndex	not-accessible	current	Y
dot1qVlanFdbId	{ dot1qVlanCurrentEntry 3 }	Unsigned32	read-only	current	Y
dot1qVlanCurrentEgresPorts	{ dot1qVlanCurrentEntry 4 }	PortList	read-only	current	Y
dot1qVlanCurrentUngaggedPorts	{ dot1qVlanCurrentEntry 5 }	PortList	read-only	current	Y
dot1qVlanStatus	{ dot1qVlanCurrentEntry 6 }	INTEGER other (1) permanent (2) dynamicGvrp (3)	read-only	current	Y
dot1qVlanCreationTime	{ dot1qVlanCurrentEntry 7 }	TimeTicks	read-only	current	Y

5.17.4.4.2.1 dot1qVlanTimeMark

A TimeFilter [Section 4.8, Page 13] for this entry. See the TimeFilter textual convention to see how this works.

Valid Object Value(s)

This object must be zero when indexing into this table.

5.17.4.4.2.2 dot1qVlanIndex

The VLAN-ID or other identifier referring to this VLAN.

5.17.4.4.2.3 dot1qVlanFdbId

The Filtering Database used by this VLAN. This is one of the dot1qFdbId values in the dot1qFdbTable. This value is allocated automatically by the device whenever the VLAN is created: either dynamically by GVRP, or by

management, in dot1qVlanStaticTable (Section 5.17.4.4.3, Page 292). Allocation of this value follows the learning constraints defined for this VLAN in dot1qLearningConstraintsTable (Section 5.17.4.4.8, Page 297).

5.17.4.4.2.4 dot1qVlanCurrentEgressPorts

The set of ports which are transmitting traffic for this VLAN as either tagged or untagged frames.

5.17.4.4.2.5 dot1qVlanCurrentUntaggedPorts

The set of ports which are transmitting traffic for this VLAN as untagged frames.

5.17.4.4.2.6 dot1qVlanStatus

This object indicates the status of this entry.

- other(1) - this entry is currently in use but the conditions under which it will remain so differ from the following values.
- permanent(2) - this entry, corresponding to an entry in dot1qVlanStaticTable (Section 5.17.4.4.3, Page 292), is currently in use and will remain so after the next reset of the device. The port lists for this entry include ports from the equivalent dot1qVlanStaticTable entry and ports learnt dynamically.
- dynamicGvrp(3) - this entry is currently in use and will remain so until removed by GVRP. There is no static entry for this VLAN and it is removed when the last port leaves the VLAN.

5.17.4.4.2.7 dot1qVlanCreationTime

The value of sysUpTime when this VLAN was created.

5.17.4.4.3 dot1qVlanStaticTable

The Static VLAN Database is a table containing static configuration information for each VLAN configured into the device by (local or network) management. All entries are permanent and are restored after the device is reset.

Each entry consists of static information for a VLAN configured into the device by (local or network) management.

Index:

- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)

OIDs:

- dot1qVlanStaticTable - { dot1qVlan 3 } 1.3.6.1.2.1.17.7.1.4.3
- dot1qVlanStaticEntry - { dot1qVlanStaticTable 1 } 1.3.6.1.2.1.17.7.1.4.3.1

Table 178. dot1qVlanStaticTable

Object	OID	Syntax	Access	Status	Supported
dot1qVlanStaticName	{ dot1qVlanStaticEntry 1 }	SnmpAdminString (SIZE (0…32))	read-create	current	Y
dot1qVlanStaticEgressPorts	{ dot1qVlanStaticEntry 2 }	PortList	read-create	current	Y
dot1qVlanForbiddenEgressPorts	{ dot1qVlanStaticEntry 3 }	PortList	read-create	current	Y
dot1qVlanStaticUntaggedPorts	{ dot1qVlanStaticEntry 4 }	PortList	read-create	current	Y
dot1qVlanStaticRowStatus	{ dot1qVlanStaticEntry 5 }	RowStatus	read-create	current	Y

5.17.4.4.3.1 dot1qVlanStaticName

An administratively assigned string, which may be used to identify the VLAN.

5.17.4.4.3.2 dot1qVlanStaticEgressPorts

The set of ports which are permanently assigned to the egress list for this VLAN by management. Changes to a bit in this object affect the per-port per-VLAN Registrar control for Registration Fixed for the relevant GVRP state machine on each port. A port may not be added in this set if it is already a member of the set of ports in dot1qVlanForbiddenEgressPorts. The default value of this object is a string of zeros of appropriate length, indicating not fixed.

5.17.4.4.3.3 dot1qVlanForbiddenEgressPorts

The set of ports which are prohibited by management from being included in the egress list for this VLAN. Changes to this object that cause a port to be included or excluded affect the per-port per-VLAN Registrar control for Registration Forbidden for the relevant GVRP state machine on each port. A port may not be added in this set if it is already a member of the set of ports in dot1qVlanStaticEgressPorts. The default value of this object is a string of zeros of appropriate length, excluding all ports from the forbidden set.

5.17.4.4.3.4 dot1qVlanStaticUntaggedPorts

The set of ports which should transmit egress packets for this VLAN as untagged. The default value of this object for the default VLAN (dot1qVlanIndex = 1) is a string of appropriate length including all ports. There is no specified default for other VLANs. If a device agent cannot support the set of ports being set then it will reject the set operation with an error. An example might be if a manager attempts to set more than one VLAN to be untagged on egress where the device does not support this IEEE 802.1Q option.

5.17.4.4.3.5 dot1qVlanStaticRowStatus

This object indicates the status of this entry.

5.17.4.4.4 dot1qNextFreeLocalVlanIndex

The next available value for dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291) of a local VLAN entry in dot1qVlanStaticTable (Section 5.17.4.4.3, Page 292). This will report values >= 4096 if a new Local VLAN may be created or else the value 0 if this is not possible. A row creation operation in this table for an entry with a local VlanIndex value may fail if the current value of this object is not used as the index. Even if the value read is used, there is no guarantee that it will still be the valid index when the create operation is attempted - another manager may have already got in during the intervening time interval. In this case, dot1qNextFreeLocalVlanIndex (Section 5.17.4.4.4, Page 293) should be re-read and the creation re-tried with the new value.

This value will automatically change when the current value is used to create a new row.

5.17.4.4.5 dot1qPortVlanTable

The VLAN Port Configuration table contains per port control and status information for VLAN configuration in the device.

Each entry consists of information controlling VLAN configuration for a port on the device.

Augments:

- dot1dBasePortTable(Section 5.15.4.1.4, Page 259)

Index:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)

OIDs:

- dot1qPortVlanTable - { dot1qVlan 5 } 1.3.6.1.2.1.17.7.1.4.5
- dot1qPortVlanEntry - { dot1qPortVlanTable 1 } 1.3.6.1.2.1.17.7.1.4.5.1

Table 179. dot1qPortVlanTable

Object	OID	Syntax	Access	Status	Supported
dot1qPvid	{ dot1qPortVlanEntry 1 }	VlanIndex	read-write	current	Y
dot1qPortAcceptableFrameTypes	{ dot1qPortVlanEntry 2 }	INTEGER AdmitAll (1) AdmitOnlyVlanTagged (2)	read-write	current	Y
dot1qPortIngressFiltering	{ dot1qPortVlanEntry 3 }	TruthValue	read-write	current	Y
dot1qPortGvrpStatus	{ dot1qPortVlanEntry 4 }	EnabledStatus	read-write	current	N
dot1qPortGvrpFailedRegistrations	{ dot1qPortVlanEntry 5 }	EnableStatus	read-write	current	N
dot1qPortGvrpLastPduOrigin	{ dot1qPortVlanEntry 6 }	MacAddress	read-only	current	N

5.17.4.4.5.1 dot1qPvid

The PVID, the VLAN ID assigned to untagged frames or Priority-Tagged frames received on this port. The default value for this object is 1.

5.17.4.4.5.2 dot1qPortAcceptableFrameTypes

- admitAll(1) - untagged frames or Priority-Tagged frames received on this port are accepted and assigned to the PVID for this port.
- admitOnlyVlanTagged(2) - the device will discard untagged frames or Priority-Tagged frames received on this port.

This control does not affect VLAN independent BPDU frames, such as GVRP and STP. It does affect VLAN dependent BPDU frames, such as GMRP.

The default value for this object is admitAll (1).

5.17.4.4.5.3 dot1qPortIngressFiltering

- true(1) - the device will discard incoming frames for VLANs which do not include this Port in its Member set.
- false(2) - the port will accept all incoming frames.

This control does not affect VLAN independent BPDU frames, such as GVRP and STP. It does affect VLAN dependent BPDU frames, such as GMRP.

The default value is false.

5.17.4.4.5.4 dot1qPortGvrpStatus

The state of GVRP operation on this port. The value enabled(1) indicates that GVRP is enabled on this port, as long as dot1qGvrpStatus (Section 5.17.4.1.5, Page 282) is also enabled for this device. When disabled(2) but dot1qGvrpStatus is still enabled for the device, GVRP is disabled on this port: any GVRP packets received are silently discarded and no GVRP registrations are propagated from other ports. This object affects all GVRP Applicant and Registrar state machines on this port. A transition from disabled(2) to enabled(1) will cause a reset of all GVRP state machines on this port.

The default value is enabled.

5.17.4.4.5.5 dot1qPortGvrpFailedRegistrations

The total number of failed GVRP registrations, for any reason, on this port.

5.17.4.4.5.6 dot1qPortGvrpLastPduOrigin

The Source MAC Address of the last GVRP message received on this port.

5.17.4.4.6 dot1qPortVlanStatisticsTable

The Per-port VLAN Statistics Table containing per-port, per-VLAN statistics for traffic received. Separate objects are provided for both the most significant and least-significant bits of statistics counters for ports that are associated with this transparent bridge. The most-significant bit objects are only required on high capacity interfaces, as defined in the conformance clauses for these objects. This mechanism is provided as a way to read 64-bit counters for agents which support only SNMPv1.

Note that the reporting of most-significant and least-significant counter bits separately runs the risk of missing an overflow of the lower bits in the interval between sampling. The manager must be aware of this possibility, even within the same varbindlist, when interpreting the results of a request or asynchronous notification.

Each entry consists of Traffic statistics for a VLAN on an interface.

Indices:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)
- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)

OIDs:

- dot1qPortVlanStatisitcsTable - { dot1qVlan 6 } 1.3.6.1.2.1.17.7.1.4.6
- dot1qPortVlanStatisticsEntry - { dot1qPortVlanStatisitcsTable 1 } 1.3.6.1.2.1.17.7.1.4.6.1

Table 180. dot1qPortVlanStatisticsTable

Object	OID	Syntax	Access	Status	Supported
dot1qTpVlanPortInFrames	{ dot1qPortVlanStatisticsEntry 1 }	Counter32	read-write	current	Y
dot1qTpVlanPortOutFrames	{ dot1qPortVlanStatisticsEntry 2 }	Counter32	read-only	current	Y
dot1qTpVlanPortInDiscards	{ dot1qPortVlanStatisticsEntry 3 }	Counter32	read-only	current	Y
dot1qTpVlanPortInOverflowFrames	{ dot1qPortVlanStatisticsEntry 4 }	Counter32	read-only	current	Y
dot1qTpVlanPortOutOverflowFrames	{ dot1qPortVlanStatisticsEntry 5 }	Counter32	read-only	current	Y
dot1qTpVlanPortInOverflowDiscards	{ dot1qPortVlanStatisticsEntry 6 }	Counter32	read-only	current	Y

5.17.4.4.6.1 dot1qTpVlanPortInFrames

The number of valid frames received by this port from its segment which were classified as belonging to this VLAN. Note that a frame received on this port is counted by this object if and only if it is for a protocol being processed by the local forwarding process for this VLAN. This object includes received bridge management frames classified as belonging to this VLAN (such as GMRP, but not GVRP or STP).

5.17.4.4.6.2 dot1qTpVlanPortOutFrames

The number of valid frames transmitted by this port to its segment from the local forwarding process for this VLAN. This includes bridge management frames originated by this device which are classified as belonging to this VLAN (such as GMRP, but not GVRP or STP).

5.17.4.4.6.3 dot1qTpVlanPortInDiscards

The number of valid frames received by this port from its segment which were classified as belonging to this VLAN which were discarded due to VLAN related reasons. Specifically, the IEEE 802.1Q counters for Discard Inbound and Discard on Ingress Filtering.

5.17.4.4.6.4 dot1qTpVlanPortInOverflowFrames

The number of times the associated dot1qTpVlanPortInFrames (Section 5.17.4.4.6.1, Page 295) counter has overflowed.

5.17.4.4.6.5 dot1qTpVlanPortOutOverflowFrames

The number of times the associated dot1qTpVlanPortOutFrames (Section 5.17.4.4.6.2, Page 295) counter has overflowed.

5.17.4.4.6.6 dot1qTpVlanPortInOverflowDiscards

The number of times the associated dot1qTpVlanPortInDiscards (Section 5.17.4.4.6.3, Page 296) counter has overflowed.

5.17.4.4.7 dot1qPortVlanHCStatisticsTable

Port VLAN High Capacity Statistics Table contains per port, per VLAN statistics for traffic on high capacity interfaces.

Each entry consists of traffic statistics for a VLAN on a high capacity interface.

Indices:

- dot1dBasePort (Section 5.15.4.1.4.1, Page 259)
- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)

OIDs:

- dot1qPortVlanHCStatistics - { dot1qVlan 7 } 1.3.6.1.2.1.17.7.1.4.7
- dot1qPortVlanHCStatisticsEntry - { dot1qPortVlanHCStatistics 1 } 1.3.6.1.2.1.17.7.1.4.7.1

Table 181. dot1qPortVlanHCStatisticsTable

Object	OID	Syntax	Access	Status	Supported
dot1qTpVlanPortHCInFrames	{ dot1qPortVlanHCStatisticsEntry 1 }	Counter64	read-only	current	N
dot1qTpVlanPortHCOutFrames	{ dot1qPortVlanHCStatisticsEntry 2 }	Counter64	read-only	current	N
dot1qTpVlanPortHCInDiscards	{ dot1qPortVlanHCStatisticsEntry 3 }	Counter64	read-only	current	N

5.17.4.4.7.1 dot1qTpVlanPortHCInFrames

The number of valid frames received by this port from its segment which were classified as belonging to this VLAN. Note that a frame received on this port is counted by this object if and only if it is for a protocol being processed by the local forwarding process for this VLAN. This object includes received bridge management frames classified as belonging to this VLAN (such as GMRP, but not GVRP or STP).

5.17.4.4.7.2 dot1qTpVlanPortHCOutFrames

The number of valid frames transmitted by this port to its segment from the local forwarding process for this VLAN. This includes bridge management frames originated by this device which are classified as belonging to this VLAN (such as GMRP, but not GVRP or STP).

5.17.4.4.7.3 dot1qTpVlanPortHCInDiscards

The number of valid frames received by this port from its segment which were classified as belonging to this VLAN which were discarded due to VLAN related reasons. Specifically, the IEEE 802.1Q counters for Discard Inbound and Discard on Ingress Filtering.

5.17.4.4.8 dot1qLearningConstraintsTable

The VLAN Learning Constraints Table containing learning constraints for sets of Shared and Independent VLANs.

Each entry consists of a learning constraint defined for a VLAN.

Index:

- dot1qConstraintVlan
- dot1qConstraintSet

OIDs:

- dot1qLearningConstraintsTable - { dot1qVlan 8 } 1.3.6.1.2.1.17.7.1.4.8
- dot1qLearningConstraintsEntry - { dot1qLearningConstraintsTable 1 } 1.3.6.1.2.1.17.7.1.4.8.1

Table 182. dot1qLearningConstraintsTable

Object	OID	Syntax	Access	Status	Supported
dot1qConstraintVlan	{ dot1qLearningConstraintsEntry 1 }	VlanIndex	not-accessible	current	N
dot1qConstraintSet	{ dot1qLearningConstraintsEntry 2 }	INTEGER (0…65535)	not-accessible	current	N
dot1qConstraintType	{ dot1qLearningConstraintsEntry 3 }	INTEGER independent (1) shared (2)	read-only	current	N
dot1qConstraintStatus	{ dot1qLearningConstraintsEntry 4 }	RowStatus	read-only	current	N

5.17.4.4.8.1 dot1qConstraintVlan

The index of the row in dot1qVlanCurrentTable for the VLAN constrained by this entry.

5.17.4.4.8.2 dot1qConstraintSet

The identity of the constraint set to which dot1qConstraintVlan belongs. These values may be chosen by the management station.

5.17.4.4.8.3 dot1qConstraintType

The type of constraint this entry defines.

- independent(1) - the VLAN, dot1qConstraintVlan, uses an independent filtering database from all other VLANs in the same set, defined by dot1qConstraintSet.
- shared(2) - the VLAN, dot1qConstraintVlan, shares the same filtering database as all other VLANs in the same set, defined by dot1qConstraintSet.

5.17.4.4.8.4 dot1qConstraintStatus

The status of this entry.

5.17.4.4.9 dot1qConstraintSetDefault

The identity of the constraint set to which a VLAN belongs, if there is not an explicit entry for that VLAN in dot1qLearningConstraintsTable.

5.17.4.4.10 dot1qConstraintTypeDefault

The type of constraint set to which a VLAN belongs, if there is not an explicit entry for that VLAN in dot1qLearningConstraintsTable (Section 5.17.4.4.8, Page 297). The types are as defined for dot1qConstraintType.

5.17.5 Conformance Information

5.17.5.1 Compliance

5.17.5.1.1 qBridgeCompliance

The compliance statement for device support of Virtual LAN Bridge services.

Mandatory Groups:

- qBridgeBaseGroup
- qBridgeVlanGroup
- qBridgeVlanStaticGroup
- qBridgePortGroup

Optional Groups:

- qBridgeFdbUnicastGroup - This group is mandatory for bridges that implement 802.1Q transparent bridging.
- qBridgeFdbMulticastGroup - This group is mandatory for bridges that implement 802.1Q transparent bridging.
- qBridgeServiceRequirementsGroup - This group is mandatory for bridges that implement extended filtering services. All objects must be read-write if extended-filtering services are enabled.
- qBridgeFdbStaticGroup - This group is optional.
- qBridgeVlanStatisticsGroup - This group is optional as there may be significant implementation cost associated with its support.
- qBridgeVlanStatisticsOverflowGroup - This group is optional as there may be significant implementation cost associated with its support. It is most relevant for high capacity interfaces where the SNMP agent supports only SNMPv1.
- qBridgeVlanHCStatisticsGroup - This group is optional as there may be significant implementation cost associated with its support. It is most relevant for high capacity interfaces.
- qBridgeLearningConstraintsGroup - This group is mandatory for devices implementing both Independent VLAN Learning (IVL) and Shared VLAN Learning (SVL) modes of operation of the filtering database, as defined by IEEE 802.1Q.
- qBridgeLearningConstraintDefaultGroup - This group is mandatory for devices implementing both Independent VLAN Learning (IVL) and Shared VLAN Learning (SVL) modes of operation of the filtering database, as defined by IEEE 802.1Q.

Objects that have a MIN-ACCESS defined:

- dot1qPortAcceptableFrameTypes - MIN-ACCESS read-only
  Write access is not required as this is an optional capability in IEEE 802.1Q.
- dot1qPortIngressFiltering - MIN-ACCESS read-only
  Write access is not required as this is an optional capability in IEEE 802.1Q.

- dot1qConstraintSetDefault - MIN-ACCESS read-only
  Write access is not required as this is an optional capability in IEEE 802.1Q.
- dot1qConstraintTypeDefault - MIN-ACCESS read-only
  Write access is not required as this is an optional capability in IEEE 802.1Q.

5.17.5.2 Conformance

Table 183. Q-BRIDGE-MIB Conformance (1 of 3)

Group	Description	Objects	Supported
qBridgeBaseGroup	A collection of objects providing device level control and status information for the Virtual LAN bridge services.	dot1qVlanVersionNumber	Y
		dot1qMaxVlanId	Y
		dot1qMaxSupportedVlans	Y
		dot1qNumVlans	Y
		dot1qGvrpStatus	N
qBridgeFdbUnicastGroup	A collection of objects providing information about all unicast addresses, learnt dynamically or statically configured by management, in each Filtering Database.	dot1qFdbDynamicCount	N
		dot1qTpFdbPort	Y
		dot1qTpFdbStatus	Y
qBridgeFdbMulticastGroup	A collection of objects providing information about all multicast addresses, learnt dynamically or statically configured by management, in each Filtering Database.	dot1qTpGroupEgressPorts	N
		dot1qTpGroupLearnt	N
qBridgeServiceRequirementsGroup	A collection of objects providing information about service requirements, learnt dynamically or statically configured by management, in each Filtering Database.	dot1qForwardAllPorts	N
		dot1qForwardAllStaticPorts	N
		dot1qForwardAllForbidden	N
		dot1qForwardUnregisteredPorts	N
		dot1qForwardUnregisteredStaticPorts	N
		dot1qForwardUnregisteredForbiddenPorts	N

Table 183. Q-BRIDGE-MIB Conformance (2 of 3)

Group	Description	Objects	Supported
qBridgeFdbStaticGroup	A collection of objects providing information about unicast and multicast addresses statically configured by management, in each Filtering Database or VLAN.	dot1qStaticUnicastAllowedToGoTo	Y
		dot1qStaticUnicastStatus	Y
		dot1qStaticMulticastStaticEgressPorts	N
		dot1qStaticMulticastForbiddenEgressPorts	N
		dot1qStaticMulticastStatus	N
qBridgeVlanGroup	A collection of objects providing information about unicast and multicast addresses statically configured by management, in each Filtering Database or VLAN.	dot1qVlanNumDeletes	N
		dot1qVlanFdbId	Y
		dot1qVlanCurrentEgressPorts	Y
		dot1qVlanCurrentUntaggedPorts	Y
		dot1qVlanStatus	Y
		dot1qVlanCreationTime	Y
qBridgeVlanStaticGroup	A collection of objects providing information about VLANs statically configured by management.	dot1qVlanStaticName	Y
		dot1qVlanStaticEgressPorts	Y
		dot1qVlanForbiddenEgressPorts	Y
		dot1qVlanStaticUntaggedPorts	Y
		dot1qVlanStaticRowStatus	Y
		dot1qNextFreeLocalVlanIndex	Y
qBridgePortGroup	A collection of objects providing port level VLAN control and status information for all ports.	dot1qPvid	Y
		dot1qPortAcceptableFrameTypes	Y
		dot1qPortIngressFiltering	Y
		dot1qPortGvrpStatus	N
		dot1qPortGvrpFailedRegistrations	N
		dot1qPortGvrpLastPduOrigin	N
qBridgeVlanStatisticsGroup	A collection of objects providing per-port packet statistics for all VLANs currently configured on this device.	dot1qTpVlanPortInFrames	N
		dot1qTpVlanPortOutFrames	N
		dot1qTpVlanPortInDiscards	N

Table 183. Q-BRIDGE-MIB Conformance (3 of 3)

Group	Description	Objects	Supported
qBridgeVlanStatisticsOverflowGroup	A collection of objects providing overflow counters for per-port packet statistics for all VLANs currently configured on this device for high capacity interfaces, defined as those that have the value of the corresponding instance of ifSpeed greater than 650,000,000 bits/second.	dot1qTpVlanPortInOverflowFrames	N
		dot1qTpVlanPortOutOverflowFrames	N
		dot1qTpVlanPortInOverflowDiscards	N
qBridgeVlanHCStatisticsGroup	A collection of objects providing per-port packet statistics for all VLANs currently configured on this device for high capacity interfaces, defined as those that have the value of the corresponding instance of ifSpeed greater than 650,000,000 bits/second.	dot1qTpVlanPortHCInFrames	N
		dot1qTpVlanPortHCOutFrames	N
		dot1qTpVlanPortHCInDiscards	N
qbridgeLearningConstraintsGroup	A collection of objects defining the Filtering Database constraints all VLANs have with each other.	dot1qConstraintType	N
		dot1qConstraintTypeDefault	N
qBridgeLearningConstraintDefaultGroup	A collection of objects defining the default Filtering Database constraints for VLANs which have no specific constraints defined.	dot1qConstraintSetDefault	N
		dot1qConstraintTypeDefault	N

5.18 IGMP-STD-MIB — RFC 2933

5.18.1 Introduction

The MIB module for IGMP Management.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200009280000Z,” or September 28, 2000.

5.18.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.18.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- igmpStdMIB - { mib-2 85 } 1.3.6.1.2.1.85
- igmpMIBObjects - { igmpStdMIB 1 } 1.3.6.1.2.1.85.1

5.18.4 MIB Objects

Table 184. IGMP-STD-MIB Objects

Object	OID	Syntax	Access	Status	Supported
igmpInterfaceTable	{ igmpMIBObjects 1 }	SEQUENCE of igmpInterfaceEntry	not-accessible	current	Y
igmpCacheTable	{ igmpMIBObjects 2 }	SEQUENCE of igmpCacheEntry	not-accessible	current	Y

5.18.4.1 igmpInterfaceTable

The (conceptual) table listing the interfaces on which IGMP is enabled.

Note:This table is not implemented as a dynamic table; rows can’t be created or destroyed. A row in this table will exist for each and every DSL interface and potential Ethernet interface that can carry multicast data.

Index:

- igmpInterfaceIfIndex

OIDs:

- igmpInterfaceTable - { igmpMIBObjects 1 } 1.3.6.1.2.1.85.1.1
- igmpInterfaceEntry - { igmpInterfaceTable 1 } 1.3.6.1.2.1.85.1.1.1

Table 185. igmpInterfaceTable

Object	OID	Syntax	Access	Status	Supported
igmpInterfaceIfIndex	{ igmpInterfaceEntry 1 }	InterfaceIndex	not-accessible	current	Y
igmpInterfaceQueryInterval	{ igmpInterfaceEntry 2 }	Unsigned32	read-write[a]	current	Y
igmpInterfaceStatus	{ igmpInterfaceEntry 3 }	RowStatus	read-write[a]	current	Y
igmpInterfaceVersion	{ igmpInterfaceEntry 4 }	Unsigned32	read-write[a]	current	Y
igmpInterfaceQuerier	{ igmpInterfaceEntry 5 }	IpAddress	read-only	current	Y
igmpInterfaceQueryMaxResponseTime	{ igmpInterfaceEntry 6 }	Unsigned32(0…255)	read-write[a]	current	Y
igmpInterfaceQuerierUpTime	{ igmpInterfaceEntry 7 }	TimeTicks	read-only	current	N
igmpInterfaceQuerierExpiryTime	{ igmpInterfaceEntry 8 }	TimeTicks	read-only	current	N
igmpInterfaceVersion1QuerierTimer	{ igmpInterfaceEntry 9 }	TimeTicks	read-only	current	Y
igmpInterfaceWrongVersionQueries	{ igmpInterfaceEntry 10 }	Counter32	read-only	current	Y
igmpInterfaceJoins	{ igmpInterfaceEntry 11 }	Counter32	read-only	current	Y
igmpInterfaceProxyIfIndex	{ igmpInterfaceEntry 12 }	InterfaceIndexOrZero	read-write[a]	current	Y
igmpInterfaceGroups	{ igmpInterfaceEntry 13 }	Gauge32	read-only	current	Y
igmpInterfaceRobustness	{ igmpInterfaceEntry 14 }	Unsigned32(1…10)[b]	read-write[a]	current	Y
igmpInterfaceLastMembQueryIntvl	{ igmpInterfaceEntry 15 }	Unsigned32(0…255)	read-write[a]	current	Y

a. Product imposed restriction. This table is not dynamic, therefore rows can't be created.

b. Product imposed restriction.

5.18.4.1.1 igmpInterfaceIfIndex

The ifIndex (Section 5.3.4.3.1, Page 37) value of the interface for which IGMP is enabled.

5.18.4.1.2 igmpInterfaceQueryInterval

The frequency at which IGMP Host-Query packets are transmitted on this interface in units of seconds.

DEFVAL-125 (Packet transmitted every 125 seconds)

5.18.4.1.3 igmpInterfaceStatus

The activation of a row enables IGMP on the interface. The destruction of a row disables IGMP on the interface.

Valid Object Value(s)

The following values are supported to allow entries to be modified:

- active(1)

5.18.4.1.4 igmpInterfaceVersion

The version of IGMP which is running on this interface. This object can be used to configure a router capable of running either value. For IGMP to function correctly, all routers on a LAN must be configured to run the same version of IGMP on that LAN.

DEFVAL-2

5.18.4.1.5 igmpInterfaceQuerier

The address of the IGMP Querier on the IP subnet to which this interface is attached.

5.18.4.1.6 igmpInterfaceQueryMaxResponseTime

The maximum query response time advertised in IGMPv2 queries on this interface.

The units are in tenths of a second. The valid range of this object is 0 to 25.5 seconds.

DEFVAL-100 (10.0 seconds)

5.18.4.1.7 igmpInterfaceQuerierUpTime

The time since igmpInterfaceQuerier was last changed.

5.18.4.1.8 igmpInterfaceQuerierExpiryTime

The amount of time remaining before the Other Querier Present Timer expires. If the local system is the querier, the value of this object is zero.

5.18.4.1.9 igmpInterfaceVersion1QuerierTimer

The time remaining until the host assumes that there are no IGMPv1 routers present on the interface. While this is non-zero, the host will reply to all queries with version 1 membership reports.

5.18.4.1.10 igmpInterfaceWrongVersionQueries

The number of queries received whose IGMP version does not match igmpInterfaceVersion (Section 5.18.4.1.4, Page 304), over the lifetime of the row entry. IGMP requires that all routers on a LAN be configured to run the same version of IGMP. Thus, if any queries are received with the wrong version, this indicates a configuration error.

5.18.4.1.11 igmpInterfaceJoins

The number of times a group membership has been added on this interface; that is, the number of times an entry for this interface has been added to the Cache Table. This object gives an indication of the amount of IGMP activity over the lifetime of the row entry.

5.18.4.1.12 igmpInterfaceProxyIfIndex

Some devices implement a form of IGMP proxying whereby memberships learned on the interface represented by this row, cause IGMP Host Membership Reports to be sent on the interface whose ifIndex value is given by this object. Such a device would implement the igmpV2RouterMIBGroup only on its router interfaces (those interfaces with non-zero igmpInterfaceProxyIfIndex). Typically, the value of this object is 0, indicating that no proxying is being done.

DEFVAL-0 (No proxying is being done)

5.18.4.1.13 igmpInterfaceGroups

The current number of entries for this interface in the Cache Table (Section 5.18.4.2, Page 306).

5.18.4.1.14 igmpInterfaceRobustness

The Robustness Variable allows tuning for the expected packet loss on a subnet. If a subnet is expected to be lossy, the Robustness Variable may be increased. IGMP is robust to (Robustness Variable-1) packet losses.

The valid range for this object is 1 to 10 per product imposed restriction

DEFVAL-2

5.18.4.1.15 igmpInterfaceLastMembQueryIntvl

The Last Member Query Interval is the Max Response Time inserted into Group-Specific Queries sent in response to Leave Group messages, and is also the amount of time between Group-Specific Query messages. This value may be tuned to modify the leave latency of the network. A reduced value results in reduced time to detect the loss of the last member of a group. The value of this object is irrelevant if igmpInterfaceVersion is 1.

The units are in tenths of a second. The valid range for this object is 0 to 25.5 seconds.

DEFVAL-1 (0.1 second)

5.18.4.2 igmpCacheTable

The (conceptual) table listing the IP multicast groups for which there are members on a particular interface.

Note:This table is not implemented as a dynamic table; rows can't be created or destroyed. A row in this table will exist for each and every multicast stream on every interface that is carrying multicast data where the "join" has been snooped.

Indicies:

- igmpCacheAddress
- igmpCacheIfIndex

OIDs:

- igmpCacheTable - { igmpMIBObjects 2 } 1.3.6.1.2.1.85.1.2
- igmpCacheEntry - { igmpCacheTable 1 } 1.3.6.1.2.1.85.1.2.1

Table 186. igmpCacheTable

Object	OID	Syntax	Access	Status	Supported
igmpCacheAddress	{ igmpCacheEntry 1 }	IpAddress	not-accessible	current	Y
igmpCacheIfIndex	{ igmpCacheEntry 2 }	InterfaceIndex	not-accessible	current	Y
igmpCacheSelf	{ igmpCacheEntry 3 }	TruthValue	read-only[a]	current	N
igmpCacheLastReporter	{ igmpCacheEntry 4 }	IpAddress	read-only	current	N
igmpCacheUpTime	{ igmpCacheEntry 5 }	TimeTicks	read-only	current	N
igmpCacheExpiryTime	{ igmpCacheEntry 6 }	TimeTicks	read-only	current	N
igmpCacheStatus	{ igmpCacheEntry 7 }	RowStatus	read-only[a]	current	Y
igmpCacheVersion1HostTimer	{ igmpCacheEntry 8 }	TimeTicks	read-only	current	N

a. Product imposed restriction.

5.18.4.2.1 igmpCacheAddress

The IP multicast group address for which this entry contains information.

5.18.4.2.2 igmpCacheIfIndex

The interface for which this entry contains information for an IP multicast group address.

5.18.4.2.3 igmpCacheSelf

An indication of whether the local system is a member of this group address on this interface.

DEFVAL-true

5.18.4.2.4 igmpCacheLastReporter

The IP address of the source of the last membership report received for this IP Multicast group address on this interface. If no membership report has been received, this object has the value 0.0.0.0.

5.18.4.2.5 igmpCacheUpTime

The time elapsed since this entry was created.

5.18.4.2.6 igmpCacheExpiryTime

The minimum amount of time remaining before this entry is aged out. A value of 0 indicates that the entry is only present because igmpCacheSelf is true and that if the router left the group, this entry would be aged out immediately. Note that some implementations may process membership reports from the local system in the same way as reports from other hosts, so a value of 0 is not required.

5.18.4.2.7 igmpCacheStatus

The status of this entry.

Valid Object Value(s)

- active(1)

5.18.4.2.8 igmpCacheVersion1HostTimer

The time remaining until the local router will assume that there are no longer any IGMP version 1 members on the IP subnet attached to this interface. Upon hearing any IGMPv1 Membership Report, this value is reset to the group membership timer. While this time remaining is non-zero, the local router ignores any IGMPv2 Leave messages for this group that it receives on this interface.

5.18.5 Conformance Information

5.18.5.1 Compliance

5.18.5.1.1 igmpV1HostMIBCompliance

The compliance statement for hosts running IGMPv1 and implementing the IGMP MIB.

Mandatory Groups:

- igmpBaseMIBGroup

Objects that have a MIN-ACCESS defined:

- igmpInterfaceStatus - MIN-ACCESS read-only
  Write access is not required.
- igmpCacheStatus - MIN-ACCESS read-only
  Write access is not required.

5.18.5.1.2 igmpV1RouterMIBCompliance

The compliance statement for routers running IGMPv1 and implementing the IGMP MIB.

Mandatory Groups:

- igmpBaseMIBGroup
- igmpRouterMIBGroup

Objects that have a MIN-ACCESS defined:

- igmpInterfaceStatus - MIN-ACCESS read-only
  Write access is not required.
- igmpCacheStatus - MIN-ACCESS read-only
  Write access is not required.

5.18.5.1.3 igmpV2HostMIBCompliance

The compliance statement for hosts running IGMPv2 and implementing the IGMP MIB.

Mandatory Groups:

- igmpBaseMIBGroup
- igmpV2HostMIBGroup

Objects that have a MIN-ACCESS defined:

- igmpInterfaceStatus - MIN-ACCESS read-only
  Write access is not required.
- igmpCacheStatus - MIN-ACCESS read-only
  Write access is not required.

5.18.5.1.4 igmpV2RouterMIBCompliance

The compliance statement for routers running IGMPv2 and implementing the IGMP MIB.

Mandatory Groups:

- igmpBaseMIBGroup
- igmpRouterMIBGroup
- igmpV2RouterMIBGroup

Objects that have a MIN-ACCESS defined:

- igmpInterfaceStatus - MIN-ACCESS read-only
  Write access is not required.
- igmpCacheStatus - MIN-ACCESS read-only
  Write access is not required.

5.18.6 Conformance

Table 187. IGMP-STD-MIB Conformance

Group	Description	Objects	Supported
igmpBaseMIBGroup	The basic collection of objects providing management of IGMP version 1 or 2.	igmpCacheSelf	N
		igmpCacheStatus	Y
		igmpInterfaceStatus	Y
igmpRouterMIBGroup	A collection of additional objects for management of IGMP version 1 or version 2 routers.	igmpCacheUpTime	N
		igmpCacheExpiryTime	Y
		igmpInterfaceJoins	Y
		igmpInterfaceGroups	Y
		igmpCacheLastReporter	Y
		igmpInterfaceQuerierUpTime	N
		igmpInterfaceQuerierExpiryTime	N
		igmpInterfaceQueryInterval	Y
igmpV2HostMIBGroup	A collection of additional objects for management of IGMP version 2 in hosts.	igmpInterfaceVersion1QuerierTimer	Y
igmpHostOptMIBGroup	A collection of optional objects for IGMP hosts. Supporting this group can be especially useful in an environment with a router which does not support the IGMP MIB.	igmpCacheLastReporter	Y
		igmpInterfaceQuerier	Y
igmpV2RouterMIBGroup	A collection of additional objects for management of IGMP version 2 in routers.	igmpInterfaceVersion	Y
		igmpInterfaceQuerier	Y
		igmpInterfaceQueryMaxResponseTime	Y
		igmpInterfaceRobustness	Y
		igmpInterfaceWrongVersionQueries	Y
		igmpInterfaceLastMembQueryIntvl	Y
		igmpCacheVersion1HostTimer	Y
igmpV2ProxyMIBGroup	A collection of additional objects for management of IGMP proxy devices.	igmpInterfaceProxyIfIndex	Y

5.19 PDN-MPE-DEVICE-CONTROL-MIB — mpe_Control.mib

5.19.1 Introduction

This MIB Module allows a user to reset a device in the DSLAM, as well as perform various operations related to the storage and retrieval of firmware and configuration files on devices.

The version of the MIB used for this document had a LAST-UPDATED of 0011211800Z which decodes to November 21, 2002.

5.19.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- ResetStates - This object defines the enumerations of values that can be applied to the reset object.
  - — SYNTAX INTEGER
    - noOp(1)
    - reset(2)
    - resetToFactoryDefaults(3)

5.19.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-mpe - { paradyne 12 } 1.3.6.1.4.1.1795.2.24.12

- mpeDevControl - { pdn-mpe 10 } 1.3.6.1.4.1.1795.2.24.12.10
- mpeDevControlMIBObjects - { mpeDevControl 1 } 1.3.6.1.4.1.1795.2.24.12.10.1
- mpeDevControlMIBTraps - { mpeDevControl 2 } 1.3.6.1.4.1.1795.2.24.12.10.2
- mpeDevControlMIBTrapsV2 - { mpeDevControlMIBObjects 0 } 1.3.6.1.4.1.1795.2.24.12.10.1.0
- mpeDevHwControl - { mpeDevControlMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.1
- mpeDevFileXferConfig - { mpeDevControlMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.12.10.1.2
- mpeDevFirmwareControl - { mpeDevControlMIBObjects 3 } 1.3.6.1.4.1.1795.2.24.12.10.1.3
- mpeDevTestControl - { mpeDevControlMIBObjects 4 } 1.3.6.1.4.1.1795.2.24.12.10.1.4

5.19.4 MIB Objects

Table 188. PDN-MPE-DEVICE-CONTROL-MIB Objects

Object	OID	Description	Supported
mpeDevHwControl	{ mpeDevControlMIBObjects 1 }	Hardware Control Group	Y
mpeDevFileXferConfig	{ mpeDevControlMIBObjects 2 }	File Transfer Group	N
mpeDevFirmwareControl	{ mpeDevControlMIBObjects 3 }	Firmware Control Group	N
mpeDevTestControl	{ mpeDevControlMIBObjects 4 }	Device Specific Test Group	N

5.19.4.1 mpeDevHwControl

The Hardware Control Group.

Table 189. mpeDevHwControl

Object	OID	Syntax	Access	Status	Supported
mpeDevControlTable	{ mpeDevHwControl 1 }	SEQUENCE OF mpeDevControlEntry	not-accessible	current	Y
mpeDevControlSelfTestTable	{ mpeDevHwControl 2 }	SEQUENCE OF devControlSelfTestEntry	not-accessible	current	N

5.19.4.1.1 mpeDevControlTable

A table that contains generic information about Card Control.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- mpeDevControlTable - { mpeDevHwControl 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.1.1
- mpeDevControlEntry - { mpeDevControlTable 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.1.1.1

Table 190. mpeDevControlTable

Object	OID	Syntax	Access	Status	Supported
mpeDevControlReset	{ mpeDevControlEntry 1 }	ResetStates	read-create	current	Y

5.19.4.1.1.1 mpeDevControlReset

Writing the value reset (2) to this object initiates a Hardware power-on reset of the device. Writing the value resetToFactoryDefaults (3) causes the device to re-configure itself with factory defaults. The value read from this object is noOp(1).

Valid Object Value(s)

There are two types of reset: hardware reset and software reset. A hardware reset will cause the coldStart trap to be sent while a software reset will cause a warmStart trap to be sent.

Table 191. mpeDevControlReset

Locator String	entPhysicalIndex	reset(2)
System	System	Hardware Reset
Unit	Unit	Hardware Reset

5.19.4.1.2 mpeDevControlSelfTestTable

A table containing information to control device specific Self-Test operations.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- mpeDevControlSelfTestTable - { mpeDevHwControl 2 }1.3.6.1.4.1.1795.2.24.12.10.1.1.2
- devControlSelfTestEntry - { mpeDevControlSelfTestTable 1 }1.3.6.1.4.1.1795.2.24.12.10.1.1.2.1

Table 192. mpeDevControlSelfTestTable

Object	OID	Syntax	Access	Status	Supported
mpeDevControlExtendedSelfTest	{ devControlSelfTestEntry 1 }	INTEGER noOp(1) enableExtendSelfTestAn-dReset(2)	read-write	current	N

5.19.4.1.2.1 mpeDevControlExtendedSelfTest

The following values are defined for this object:

- noOp(1)
- enableExtendSelfTestAndReset(2)

This object is used to start the extended Power-On Self-Test test. Writing the value enableExtendSelfTestAndReset(2) to this object resets the device and puts it into extended Power-On Self-Test mode. During device initialization, the extended Self-Test would be carried out first and then the device would carry out its normal boot operation. The value read from this object is noOp(1). Writing back the same value, noOp(1) will not start the test. Note that if extended Self-Test fails for some reason, after re-initialization, the value of this object will default to enableExtendSelfTestAndReset(2) again unless the manager explicitly chooses not to do so.

5.19.4.2 mpeDevFileXferConfig

The File Transfer Config Group.

Table 193. mpeDevFileXferConfig

Object	OID	Syntax	Access	Status	Supported
mpeDevFileXferConfigTable	{ mpeDevFileXferConfig 1 }	SEQUENCE OF mpeDevFileXferConfigEntry	not-accessible	current	N

5.19.4.2.1 mpeDevFileXferConfigTable

The Paradyne FileXfer Client Config Table.

Use of the File Tranfer MIB could be used with other MIBS in the following manner:

- Use another MIB/means to verify available space /make room for a file to be transferred to this device
- Use this MIB to download the file
- Use another MIB/means to select the file you want to make active if your selecting firmware for example.
- Use another MIB/means to reset the device.

A management station wishing to initiate a file transfer needs to create an entry in this table. To do so, you must first identify the entPhysicalIndex of the device you intend to do the transfer with. You should then create the associated

instance of the row status It must also, either in the same or in successive PDUs, create an instance of mpeDevFileXferFileName, mpeDevFileXferFileType, mpeDevFileXferServerIpAddress, mpeDevFileXferOperation.

It should also modify the default values for the other configuration objects if the defaults are not appropriate.

Once the appropriate instance of all the configuration objects have been created, either by an explicit SNMP set request or by default, the row status should be set to active to initiate the request. Note that this entire procedure may be initiated via a single set request which specifies a row status of createAndGo as well as specifies valid values for the non-defaulted configuration objects.

Once the MpeDevFileXferConfigEntry request has been created (that is, the mpeDevFileXferRowStatus has been made active), the entry cannot be modified - the only operation possible after this is to delete the row.

Once the request completes, the management station should retrieve the values of the status objects of interest, and should then delete the entry. In order to prevent old entries from clogging the table, entries could be aged out, but an entry will never be deleted within 5 minutes of completing.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- mpeDevFileXferConfigTable - { mpeDevFileXferConfig 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.2.1
- mpeDevFileXferConfigEntry - { mpeDevFileXferConfigTable 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.2.1.1

Table 194. mpeDevFileXferConfigTable

Object	OID	Syntax	Access	Status	Supported
mpeDevFileXferFileName	{ mpeDevFileXferConfigEntry 1 }	DisplayString	read-create	current	N
mpeDevFileXferCopyProtocol	{ mpeDevFileXferConfigEntry 2 }	INTEGER tftp(1) ftp(2)	read-create	current	N
mpeDevFileXferFileType	{ mpeDevFileXferConfigEntry 3 }	INTEGER firmware(1) config(2)	read-create	current	N
mpeDevFileXferServerIpAddress	{ mpeDevFileXferConfigEntry 4 }	IpAddress	read-create	current	N
mpeDevFileXferUserName	{ mpeDevFileXferConfigEntry 5 }	DisplayString	read-create	current	N
mpeDevFileXferUserPassword	{ mpeDevFileXferConfigEntry 6 }	DisplayString	read-create	current	N
mpeDevFileXferOperation	{ mpeDevFileXferConfigEntry 7 }	INTEGER get(1) put(2)	read-create	current	N
mpeDevFileXferPktsSent	{ mpeDevFileXferConfigEntry 8 }	Counter32	read-only	current	N
mpeDevFileXferPktsRecv	{ mpeDevFileXferConfigEntry 9 }	Counter32	read-only	current	N
mpeDevFileXferOctetsSent	{ mpeDevFileXferConfigEntry 10 }	Counter32	read-only	current	N
mpeDevFileXferOctetsRecv	{ mpeDevFileXferConfigEntry 11 }	Counter32	read-only	current	N
mpeDevFileXferOwnerString	{ mpeDevFileXferConfigEntry 12 }	OCTET STRING (SIZE(0…255))	read-create	current	N
mpeDevFileXferStatus	{ mpeDevFileXferConfigEntry 13 }	INTEGER none(1) success(2) failure(3) inprogress(4)	read-only	current	N

Table 194. mpeDevFileXferConfigTable

Object	OID	Syntax	Access	Status	Supported
mpeDevFileXferErrorStatus	{ mpeDevFileXferConfigEntry 14 }	Integer32	read-only	current	N
mpeDevFileXferSendEvent	{ mpeDevFileXferConfigEntry 15 }	INTEGER active(1) inactive(2)	read-create	current	N
mpeDevFileXferRowStatus	{ mpeDevFileXferConfigEntry 16 }	RowStatus	read-create	current	N
mpeDevFileXferXferTime	{ mpeDevFileXferConfigEntry 17 }	TimeTicks	read-only	current	N
mpeDevFileXferFileFormat	{ mpeDevFileXferConfigEntry 18 }	INTEGER ascii(1) binary(2)	read-create	current	N

5.19.4.2.1.1 mpeDevFileXferFileName

This object contains the name of the file that is to be transferred.

5.19.4.2.1.2 mpeDevFileXferCopyProtocol

The transfer protocol that should be used to copy the file across the network. If the file transfer is to occur locally on the SNMP agent, the method of transfer is left up to the implementation, and is not restricted to the protocols below.

- tftp(1)
- ftp(2)

The default value is tftp(1).

5.19.4.2.1.3 mpeDevFileXferFileType

Specifies the type of file your want to transfer. The following values are defined for this object:

- firmware(1)
- config(2)

5.19.4.2.1.4 mpeDevFileXferServerIpAddress

This object contains the file transfer server's IP address.

5.19.4.2.1.5 mpeDevFileXferUserName

This object contains the username if a username is needed to login to the server.

5.19.4.2.1.6 mpeDevFileXferUserPassword

This object contains the password if a password is needed to login to the server. This object will return null on a read operation.

5.19.4.2.1.7 mpeDevFileXferOperation

This object contains the operation the file transfer wants to perform. The following values are defined for this object:

- get(1)
- put(2)

5.19.4.2.1.8 mpeDevFileXferPktsSent

This object contains the number of packets sent to the server at the time of interrogation.

5.19.4.2.1.9 mpeDevFileXferPktsRecv

This object contains the number of packets received from the server at the time of interrogation.

5.19.4.2.1.10 mpeDevFileXferOctetsSent

This object contains the number of octets sent to the server at the time of interrogation.

5.19.4.2.1.11 mpeDevFileXferOctetsRecv

This object contains the number of octets received from the server at the time of interrogation.

5.19.4.2.1.12 mpeDevFileXferOwnerString

The entity which currently has the “ownership” required to invoke the operation on this index.

5.19.4.2.1.13 mpeDevFileXferStatus

This object contains the status of the file transfer. The following values are defined for this object:

- none(1)
- success(2)
- failure(3)
- inprogress(4)

5.19.4.2.1.14 mpeDevFileXferErrorStatus

This object contains the reason code of the failure determined in mpeDevFileXferStatus. The reason code are specific to the file transfer protocol.

Please refer to the file transfer protocols respective RFC for clarification of the error code value meanings.

- RFC 1350 - TFTP Error Codes
- RFC 959 - FTP Error Codes

5.19.4.2.1.15 mpeDevFileXferSendEvent

This object indicates whether or not to send the mpeDevFileXferEvent event trap. The following values are defined for this object:

- active(1)
- inactive(2)

5.19.4.2.1.16 mpeDevFileXferRowStatus

This object is used to create a new row or delete an existing row in this table.

5.19.4.2.1.17 mpeDevFileXferXferTime

This object indicates the elapsed time (in hundredths of a second) of the file transfer.

5.19.4.2.1.18 mpeDevFileXferFileFormat

The following values are defined for this object:

- ascii(1)
- binary(2)

This object contains the file representation type. A file representation type of ascii implies an ASCII, non-print text file. A file representation type of binary implies an binary image file.

The default file representation type is binary(2).

5.19.4.3 mpeDevFirmwareControl

The Firmware Control Group.

Table 195. mpeDevFirmwareControl

Object	OID	Syntax	Access	Status	Supported
mpeDevFirmwareControlTable	{ mpeDevFirmwareControl 1 }	SEQUENCE OF mpeDevFirmwareControlEntry	not-accessible	current	N

5.19.4.3.1 mpeDevFirmwareControlTable

A list of the current Firmware Releases and their associated status. Each Firmware Release is indexed by a number from 1 to N. The user is able to view the Firmware Release String and Operational Status of the release (valid or invalid) and activate a valid Firmware Release by changing the Administration Status to active.

Indices:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)
- mpeDevFirmwareControlIndex

OIDs:

- mpeDevFirmwareControlTable - { mpeDevFirmwareControl 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.3.1
- mpeDevFirmwareControlEntry - { mpeDevFirmwareControlTable 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.3.1.1

Table 196. mpeDevFirmwareControlTable

Object	OID	Syntax	Access	Status	Supported
mpeDevFirmwareControlIndex	{ mpeDevFirmwareControlEntry 1 }	Integer32	read-only	current	N
mpeDevFirmwareControlRelease	{ mpeDevFirmwareControlEntry 2 }	DisplayString (SIZE (1…16))	read-only	current	N
mpeDevFirmwareControlOperStatus	{ mpeDevFirmwareControlEntry 3 }	INTEGER valid(1) invalid(2) unknown(3)	read-only	current	N
mpeDevFirmwareControlAdminStatus	{ mpeDevFirmwareControlEntry 4 }	INTEGER active(1) inactive(2)	read-create	current	N

5.19.4.3.1.1 mpeDevFirmwareControlIndex

This object is used to index the Firmware table (range 1 to N).

5.19.4.3.1.2 mpeDevFirmwareControlRelease

This object indicates the Software Release for this Firmware. If the Software Firmware is Operational Status is invalid, the Software Revision Number is blank.

5.19.4.3.1.3 mpeDevFirmwareControlOperStatus

This object indicates whether or not the indexed Firmware entry contains a valid.or invalid Firmware. The following values are defined for this object:

- valid(1)
- invalid(2)

- unknown(3)

5.19.4.3.1.4 mpeDevFirmwareControlAdminStatus

This object indicates whether or not the indexed Firmware entry is active or inactive. The following values are defined for this object:

- active(1)
- inactive(2)

Writing active(1) will activate that software release and cause the unit to reset (response may timeout). Writing active(1) to a Firmware entry whose mpeDevFirmwareControlOperStatus is invalid will return BAD VALUE. Writing inactive(2) will always return BAD VALUE.

5.19.4.4 mpeDevTestControl

The device specific test group.

Table 197. mpeDevTestControl

Object	OID	Syntax	Access	Status	Supported
mpeDevControlTestTable	{ mpeDevTestControl 3 }[a]	SEQUENCE OF mpeDevControlTestEntry	not-accessible	current	Y

a. No { mpeDevTestControl 1 } or { mpeDevTestControl 2 } objects specified in the MIB.

5.19.4.4.1 mpeDevControlTestTable

A table that contains configuration information to perform device specific tests.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- mpeDevControlTestTable - { mpeDevTestControl 3 } 1.3.6.1.4.1.1795.2.24.12.10.1.4.3
- mpeDevControlTestEntry - { mpeDevControlTestTable 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.4.3.1

Table 198. mpeDevControlTestTable

Object	OID	Syntax	Access	Status	Supported
mpeDevControlTestType	{ mpeDevControlTestEntry 1 }	INTEGER lampTest(1)	read-write	current	Y
mpeDevControlTestStatus	{ mpeDevControlTestEntry 2 }	INTEGER active(1) inactive(2)	read-only	current	Y
mpeDevControlTestCmd	{ mpeDevControlTestEntry 3 }	INTEGER start(1) stop(2)	read-write	current	Y

5.19.4.4.1.1 mpeDevControlTestType

This object is used to specify the type of the test to start. The following values are defined for this object:

- lampTest(1)

5.19.4.4.1.2 mpeDevControlTestStatus

The following values are defined for this object:

- active(1)
- inactive(2)

The test status on the device. This object indicates whether the indexed test is currently active(1) or inactive(2).

5.19.4.4.1.3 mpeDevControlTestCmd

Used to start or stop the indexed test. When read, the value returned is the next logical command. The following values are defined for this object:

- start (1)
- stop (2)

5.19.5 Traps

OIDs:

- mpeDevControlMIBTrapsV2 - { mpeDevControlMIBTraps 0 } 1.3.6.1.4.1.1795.2.24.12.10.2.0

Table 199. PDN-MPE-DEVICE-CONTROL-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
mpeDevFileXferEvent	{ mpeDevControlMIBTrapsV2 1 }	mpeDevFileXferStatus mpeDevFileXferErrorStatus mpeDevFileXferOperation mpeDevFileXferFileType mpeDevFileXferFileName	current	N

5.19.5.1 mpeDevFileXferEvent

This trap is to communicate a couple of things about the completion of a file transfer.

- mpeDevFileXferStatus - Did it complete successfully or not
- mpeDevFileXferErrorStatus - If not, what was the error code
- mpeDevFileXferOperation - What operation was performed
- mpeDevFileXferFileType - Was it a firmware xfer or config
- mpeDevFileXferFileName - The name of the file transferred

5.20 PDN-MPE-DSLAM-SYSTEM-MIB — mpe_dslam.mib

5.20.1 Introduction

The mpe dslam MIB. This MIB is written specifically to extend entPhysicalTable.

The version of the MIB used for this document had a LAST-UPDATED of 200306060000Z which decodes to June 6, 2003.

5.20.2 Textual Conventions

The following textual conventions are made in this RFC:

- MpeEntExtAdminStatus - Values to support the desired state of the entity. These values were selected to correspond to the syntax of the ifAdminStatus in the IF-MIB.
  - INTEGER
    - up(1) - The Administrator considers the entity to be in the “up” or “operational” state.
    - down(2) - The Administrator considers the entity to be in the “down” or “non-operational” state. Generally in this state, alarms or traps should not be reported.
    - testing(3) - The Administrator considers the entity to be in a state where the entity is under going some sort of tests.
- MpeEntExtOperStatus - Values to support the current operational state of the entity. These values were selected to correspond to the syntax of the ifOperStatus in the IF-MIB.
  - INTEGER
    - up(1) - The entity is operational.
    - down(2) - The entity is non-operational.
    - testing(3) - The entity is in some test mode.
    - unknown(4) - The entity is in an “unknown” state.
    - dormant(5) - The entity is in a “dormant” state.
    - notPresent(6) - The entity was present, but is currently not present.
    - reserved1(7) - Not used, but reserved.

5.20.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-mpe - { paradyne 12 } 1.3.6.1.4.1.1795.2.24.12

- mpe-dslam - { pdn-mpe 24 } 1.3.6.1.4.1.1795.2.24.12.24
- mpeSysDevDslamMIBObjects - { mpe-dslam 1 } 1.3.6.1.4.1.1795.2.24.12.24.1

- mpeSysDevDslamMIBTraps - { mpe-dslam 2 } 1.3.6.1.4.1.1795.2.24.12.24.2
- mpeEntExtAlarms - { mpeSysDevDslamMIBObjects 0 } 1.3.6.1.4.1.1795.2.24.12.24.1.1
- mpeAlarmRelay - { mpeSysDevDslamMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.12.24.1.2
- mpeEntExtMibObjects - { mpeSysDevDslamMIBObjects 3 } 1.3.6.1.4.1.1795.2.24.12.24.1.3
- mpeSysDevDslamMIBNotifications - { mpeSysDevDslamMIBTraps 1 } 1.3.6.1.4.1.1795.2.24.12.24.2.0

5.20.4 MIB Objects

Table 200. PDN-MPE-DSLAM-SYSTEM-MIB Objects

Object	OID	Description	Supported
mpeEntExtAlarms	{ mpeSysDevDslamMIBObjects 1 }	ALARM Objects	Y
mpeAlarmRelay	{ mpeSysDevDslamMIBObjects 2 }	ALARM Relay Objects	Y
mpeEntExtMibObjects	{ mpeSysDevDslamMIBObjects 3 }	Objects that extend the entPhysicalTable.	N

5.20.4.1 mpeEntExtAlarms

The Alarm Objects Group.

Table 201. mpeEntExtAlarms

Object	OID	Syntax	Access	Status	Supported
mpeEntExtAlarmTable	{ mpeEntExtAlarms 1 }	SEQUENCE OF mpeEntExtAlarmEntry	not-accessible	current	Y

5.20.4.1.1 mpeEntExtAlarmTable

This table lists the alarm states of the objects listed in the entity-MIB entPhysicalTable.

Augments:

- entPhysicalEntry

Index:

- entPhysicalIndex

OIDs:

- mpeEntExtAlarmTable - { mpeEntExtAlarms 1 } 1.3.6.1.4.1.1795.2.24.12.24.1.1.1
- mpeEntExtAlarmEntry - { mpeEntExtAlarmTable 1 } 1.3.6.1.4.1.1795.2.24.12.24.1.1.1.1

Table 202. mpeEntExtAlarmTable

Object	OID	Syntax	Access	Status	Supported
mpeEntExtAlarm	{ mpeEntExtAlarmEntry 1 }	TruthValue	read-only	current	Y

5.20.4.1.1.1 mpeEntExtAlarm

This variable indicates the alarm state of the physical entity.

5.20.4.2 mpeAlarmRelay

Alarm Relay objects. For the purpose of being used as VarBinds for the AlarmRelayInputContact trap.

Table 203. mpeAlarmRelay

Object	OID	Syntax	Access	Status	Supported
mpeAlarmRelay-EquipIndex	{ mpeAlarmRelay 1 }	Integer32	accessible-for-notify	deprecated	N
mpeAlarmRelay-InputContactState	{ mpeAlarmRelay 2 }	ContactState	accessible-for-notify	deprecated	N
mpeAlarmRelay-Table	{ mpeAlarmRelay 3 }	SEQUENCE OF mpeAlarmRelayEntry	not-accessible	current	Y

5.20.4.2.1 mpeAlarmRelayEquipIndex

This variable is the index of the equipment whose alarms are being relayed. It is a integer number starting from 1 until the max number of physical connectors the DSLAM device has to support relay.

5.20.4.2.2 mpeAlarmRelayInputContactState

This variable indicates the alarm state of the Input Contact (open/closed).

5.20.4.2.3 mpeAlarmRelayTable

This table holds objects that relate to Alarm Relay input or output. It is a sparse augments for the entPhysicalTable.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- mpeAlarmRelayTable - { mpeAlarmRelay 3 } 1.3.6.1.4.1.1795.2.24.12.24.1.2.3
- mpeAlarmRelayEntry - { mpeAlarmRelayTable 1 } 1.3.6.1.4.1.1795.2.24.12.24.1.2.3.1

Table 204. mpeAlarmRelayTable

Object	OID	Syntax	Access	Status	Supported
mpeAlarmRelayState	{ mpeAlarmRelayEntry 1 }	ContactState	read-write	current	Y

5.20.4.2.3.1 mpeAlarmRelayState

This object is used to read or write the state of an Alarm Relay input or output.

5.20.4.3 mpeEntExtMibObjects

Objects that extend the entPhysicalTable.

Table 205. mpeEntExtMibObjects

Object	OID	Syntax	Access	Status	Supported
mpeEntPhhsicalExtTable	{ mpeEntExtMibObjects 1 }	SEQUENCE OF mpeEntPhhsicalExtEntry	not-accessible	current	N

5.20.4.3.1 mpeEntPhysicalExtTable

This table contains entity information that is not defined in the standard ENTITY-MIB.

Augments:

- entPhysicalEntry

Index:

- entPhysicalIndex

OIDs:

- mpeEntPhysicalExtTable - { mpeEntExtMibObjects 1 } 1.3.6.1.4.1.1795.2.24.12.24.1.3
- mpeEntPhysicalExtEntry - { mpeEntPhysicalExtTable 1 } 1.3.6.1.4.1.1795.2.24.12.24.1.3.1

Table 206. mpeEntPhysicalExtTable

Object	OID	Syntax	Access	Status	Supported
mpeEntPhysicalExtUp-Time	{ mpeEntPhysicalExtEntry 1 }	TimeTicks	read-only	current	N
mpeEntPhysicalExt-Location	{ mpeEntPhysicalExtEntry 1 }	SnmpAdminString	read-write	current	N
mpeEntPhysicalExt-AdminStatus	{ mpeEntPhysicalExtEntry 1 }	MpeEntExtAdminStatus	read-write	current	N
mpeEntPhysicalExt-OperStatus	{ mpeEntPhysicalExtEntry 1 }	MpeEntExtOperStatus	read-only	current	N

5.20.4.3.1.1 mpeEntPhysicalExtUpTime

The time (in hundredths of a second --just like the sysUpTime) since the entity was last re-initialized.

5.20.4.3.1.2 mpeEntPhysicalExtLocation

This object allows the manager to enter the location of the physical entity where applicable. For example, in a stack, chassis or units may be in different locations. In this case, it would be logical for the manager to set this object to the location of the associated chassis.

Where the location does not make sense, it is recommended that the object return a zero-length string.

5.20.4.3.1.3 mpeEntPhysicalExtAdminStatus

The desired state of the entity.

5.20.4.3.1.4 mpeEntPhysicalExtOperStatus

The current operational state of the entity.

5.20.5 Traps

Table 207. PDN-MPE-DSLAM-SYSTEM-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
mpeCcn	{ mpeSysDevDslamMIBTraps 7 }	entPhysicalIndex	current	Y
mpeDeviceFailure	{ mpeSysDevDslamMIBTraps 15 }	entPhysicalIndex	current	N
mpeDeviceFailureCleared	{ mpeSysDevDslamMIBTraps 16 }	entPhysicalIndex	current	N
mpeNonSupportedMCC	{ mpeSysDevDslamMIBTraps 20 }	mpeSysObjectID	current	N
mpeNonSupportedChassis	{ mpeSysDevDslamMIBTraps 21 }	mpeSysObjectID	current	N
mpeAlarmRelayInputContact-StateChanged	{ mpeSysDevDslamMIBTraps 22 }	mpeAlarmRelayEquipIndex mpeAlarmRelayInputContactState	deprecated	N
mpeEntPhysicalExtEntity-Created	{ mpeSysDevDslamMIBTraps 23 }	entPhysicalIndex	current	N
mpeEntPhysicalExtEntity-Deleted	{ mpeSysDevDslamMIBTraps 24 }	entPhysicalIndex	current	N
mpeEntPhysicalExtEntity-Changed	{ mpeSysDevDslamMIBTraps 25 }	entPhysicalIndex	current	N
mpeAlarmRelayInputState-Changed	{ mpeSysDevDslamMIBTraps 26 }	mpeAlarmRelayState	current	Y

5.20.5.1 mpeCcn

This trap signifies a Configuration change or software upgrade in the xDSL card. This trap is of “warning” class.

5.20.5.2 mpeDeviceFailure

This trap signifies that the sending protocol’s device has failed and the failure was not a result of a device test. In other words, an internal device failure has been detected by the operating software for the unit.

In the BitStorm 2600 and GranDSLAM 4200 IP DSLAM, this trap indicates that an internal communication subsystem failure has occurred.

5.20.5.3 mpeDeviceFailureCleared

This trap signifies that the sending protocol’s device has failed but now it is operational. Failure was not a result of a device test. In other words, an internal device failure has been previously detected by the operating software for the unit and the failure cleared.

In the BitStorm 2600 and GranDSLAM 4200 IP DSLAM, this trap indicates that an internal communication subsystem failure cleared. The subsystem communication has been established.

5.20.5.4 mpeNonSupportedMCC

AN has detected MCC firmware release too low to support this device.

5.20.5.4.1 mpeSysObjectID

OID:

- mpeSysObjectID - { mpeSystemsEntry 2 } 1.3.6.1.4.1.1795.2.24.12.4.1.1.1.1.2

The following is an excerpt from PDN-MPE-MIB2-MIB, mpe_mib2.mib:

The vendor's authoritative identification of the network management subsystem contained in the entity. This value is allocated within the SMI enterprises subtree (1.3.6.1.4.1) and provides an easy and unambiguous means for determining "what kind of box" is being managed. For example, if vendor "Flinstones, Inc." was assigned the subtree 1.3.6.1.4.1.4242, it could assign the identifier 1.3.6.1.4.1.4242.1.1 to its "Fred Router."

- •Syntax- OBJECT IDENTIFIER
- •Access- read-only
- •Status- mandatory

5.20.5.5 mpeNonSupportedChassis

AN in slot xx has been installed in a chassis that cannot support one or more of its features.

Reference Section 5.20.5.4.1, Page 325 for detail on mpeSysObjectID.

5.20.5.6 mpeAlarmRelayInputContactStateChanged

This trap signifies that the state of the InputContact Alarm has changed since last time.

NOTE: This object has been deprecated. Please consider using the mpeAlarmRelayInputStateChanged notification.

5.20.5.7 mpeEntPhysicalExtEntityCreated

This trap signifies that an instance of this entity has been created in the entPhysicalTable.

This trap is generated in addition to the entConfigChange notification defined in RFC 2737.

5.20.5.8 mpeEntPhysicalExtEntityDeleted

This trap signifies that an instance of this entity has been deleted from the entPhysicalTable.

This trap is generated in addition to the entConfigChange notification defined in RFC 2737.

5.20.5.9 mpeEntPhysicalExtEntityChanged

This trap signifies that an instance of this entity has been modified in the entPhysicalTable as a result of a change in any of the objects in that table.

This trap is generated in addition to the entConfigChange notification defined in RFC 2737.

5.20.5.10 mpeAlarmRelayInputStateChanged

This trap signifies that the state of the Alarm Relay Input has changed.

5.21 PDN-MPE-HEALTH-AND-STATUS-MIB — mpe_HealthAndStatus.mib

5.21.1 Introduction

Enterprise Health and Status MIB (mpe support).

5.21.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.21.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-mpe - { paradyne 12 } 1.3.6.1.4.1.1795.2.24.12

- mpe-devHealth - { pdn-mpe 10 } 1.3.6.1.4.1.1795.2.24.12.10
- mpeDevHealthAndStatusMIBObjects - { mpe-devHealth 1 } 1.3.6.1.4.1.1795.2.24.12.10.1
- mpeDevHealthAndStatusMIBTraps - { mpe-devHealth 2 } 1.3.6.1.4.1.1795.2.24.12.10.2

5.21.4 MIB Objects

Table 208. PDN-MPE-HEALTH-AND-STATUS-MIB Objects

Object	OID	Syntax	Access	Status	Supported
mpeDevHealthAndStatusTable	{ mpeDevHealthAndStatusMIBObjects 1 }	SEQUENCE OF mpeDevHealthAndStatusEntry	not-accessible	mandatory	Y

5.21.4.1 mpeDevHealthAndStatusTable

A table that contains information about an Entity’s health.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- mpeDevHealthAndStatusTable - { mpeDevHealthAndStatusMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.1
- mpeDevHealthAndStatusEntry - { mpeDevHealthAndStatusTable 1 } 1.3.6.1.4.1.1795.2.24.12.10.1.1.1

Table 209. mpeDevHealthAndStatusTable

Object	OID	Syntax	Access	Status	Supported
mpeDevSelfTestResults	{ mpeDevHealthAndStatusEntry 1 }	DisplayString(SIZE(0…255))	read-only	mandatory	Y

5.21.4.1.1 mpeDevSelfTestResults

Self-test results. Self-test (or power-up test) results summarizes the test results of each CCA, where each CCA test result is separated by a semi-colon. Refer to device-specific user documentation for a complete description of the self test codes and messages.

Valid Object Value(s)

This object contains the self-test results of the device. Self-test (or power-up test) results summarizes the test results of each CCA (such as memory and CPU) where each CCA test result is separated by a semi-colon. The following are values supported for this object where %c: **P** = pass, **F** = fail, **U** = unknown status:

- Chassis - "Chassis=%c"
- Management Processor - "CPUReg=%c;CPUTimer=%c;SDRAM=%c;File System=%c"
- DSL Ports - "Memory=%c;DataPump=%c;PHY=%c"
- Ethernet Port 1 - "MAC=%c;PHY=%c;Device Reg=%c"
- Ethernet Port 2 - "MAC=%c;PHY=%c;Device Reg=%c"
- Ethernet Port 3 - "MAC=%c;PHY=%c;Device Reg=%c"

The corresponding UI elements are on the Selftest Results (Diagnostics - Selftest) Screen.

5.21.5 Traps

OIDs:

- mpeDevHealthAndStatusMIBTraps - { mpe-devHealth 2 } 1.3.6.1.4.1.1795.2.24.12.10.2

Table 210. PDN-MPE-HEALTH-AND-STATUS-MIB Traps

Trap	ENTERPRISE	:=	Variable Bindings / Objects	Supported
mpeSelfTestFailure	mpeDevHealthAndStatusMIBTraps	1	mpeDevSelfTestResults	Y

5.21.5.1 mpeSelfTestFailure

This trap signifies that the sending protocol's device has failed self test. The variable binding for this trap would be the selfTest devSelfTestResults object of the Health and Status MIB.

5.22 PDN-ARP-MIB — pdn_Arp.mib

5.22.1 Introduction

This MIB contains objects that is used configure ARP Entries and display ARP information on a Device.

The version of the MIB used for this document had a LAST-UPDATED of 0204180000Z which decodes to April 19, 2002.

5.22.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- VnidRange - The valid range for VNID IDs.

  The syntax for this object is:

  — INTEGER(2…4000)

5.22.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common- { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-arp - { pdn-common 27 } 1.3.6.1.4.1.1795.2.24.2.27
- pdnNetToMediaGenericMIBObjects - { pdn-arp 1 } 1.3.6.1.4.1.1795.2.24.2.27.1
- pdnNetToMediaMIBTraps - { pdn-arp 2 } 1.3.6.1.4.1.1795.2.24.2.27.1
- pdnNetToMediaParams - { pdnNetToMediaGenericMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.1
- pdnNetToMediaConfig - { pdnNetToMediaGenericMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.27.1.2
- pdnNetTo8023MediaConfig - { pdnNetToMediaGenericMIBObjects 3 } 1.3.6.1.4.1.1795.2.24.2.27.1.3
- pdnNetToMediaGroups - { pdnNetToMediaGenericMIBObjects 4 } 1.3.6.1.4.1.1795.2.24.2.27.1.4

- ipNetToMediaConfig - { pdnNetToMediaConfig 4 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.4

- ipNetToMediaTable - { ip 2 } 1.3.6.1.2.1.4.2
- ipNetToMediaEntry - { ipNetToMediaTable 1 } 1.3.6.1.2.1.4.2.1
- ipNetToMediaIfIndex - { ipNetToMediaEntry 1 } 1.3.6.1.2.1.4.2.1.1
- ipNetToMediaPhysAddress - { pdnNetToMediaMIBTraps 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.1

5.22.4 MIB Objects

Table 211. PDN-ARP- MIB Objects

Object	OID	Description	Supported
pdnNetToMediaParams	{ pdnNetToMediaGenericMIBObjects 1 }	ARP Parameters Configuration group	N
pdnNetToMediaConfig	{ pdnNetToMediaGenericMIBObjects 2 }	ARP Entry Configuration group	Y
pdnNetTo8023MediaConfig	{ pdnNetToMediaGenericMIBObjects 3 }	ARP Entry Configuration for 802.3 Media Cards	N
ipNetToMediaConfig	{ pdnNetToMediaConfig 4 }	Proxy ARP Configuration Group	Y
pdnNetToMediaMIBTraps	{ pdn-arp 2 }	Traps	N

5.22.4.1 pdnNetToMediaParams

This group consists of configurable ARP parameters.

OIDs:

- pdnNetToMediaParams - { pdnNetToMediaGenericMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.1

Table 212. pdnNetToMediaParams

Object	OID	Syntax	Access	Status	Supported
pdnNetToMediaParamsCompEntryTimout	{ pdnNetToMediaParams 1 }	INTEGER (0…99999)	read-write	current	N
pdnNetToMediaParamsIncompEntryTimeout	{ pdnNetToMediaParams 2 }	INTEGER (0…99999)	read-write	current	N
pdnNetToMediaParamsDefRoutEntryTimeout	{ pdnNetToMediaParams 3 }	INTEGER (0…99999)	read-write	current	N

5.22.4.1.1 pdnNetToMediaParamsCompEntryTimeout

Complete Entry Timeout is the length of time that a complete entry remains in the ARP table before removal. A complete entry is one for which there is a MAC address (that is, a node has responded to the ARP request). If an entry gets this old without being referenced, it is removed from the table. The range for this parameter is 1 to 200000 minutes.

The default value for this object is 20.

5.22.4.1.2 pdnNetToMediaParamsIncompEntryTimeout

Incomplete Entry Timeout is the length of time that an incomplete entry remains in the ARP table before removal. An incomplete entry is one for which there is no MAC address (that is, an ARP request has been made, but no response has been received.) This is also the amount of time that a packet will remain in the system while waiting for address resolution. The range for this parameter is 1 to 255 minutes.

The default value for this object is 3.

5.22.4.1.3 pdnNetToMediaParamsDefRouteEntryTimeout

Default Route Entry Timeout is the length of time a default route entry will remain in the ARP table. If default route gets this old without being referenced, an ARP request is sent to the next hop router. If no response is received, it is

removed from the ARP table and the card will switch to the next reachable default route with the highest preference. The range for this parameter is 1 to 20 minutes.

The default value for this object is 1.

5.22.4.2 pdnNetToMediaConfig

This group contains objects for configuring ARP entries.

OIDs:

- pdnNetToMediaConfig - { pdnNetToMediaGenericMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.27.1.2

Table 213. pdnNetToMediaConfig

Object	OID	Syntax	Access	Status	Supported
pdnNetToMediaConfigTable	{ pdnNetToMediaConfig 1 }	SEQUENCE of pdnNetToMediaConfigEntry	not-accessible	current	N
pdnNetToMediaClearAllArp	{ pdnNetToMediaConfig 2 }	INTEGER noop(1) clear(2)	read-write	current	Y
pdnNetToMediaProxyArpTable	{ pdnNetToMediaConfig 3 }	SEQUENCE of pdnNetToMeidaProxyArpEntry	not-accessible	current	N

5.22.4.2.1 pdnNetToMediaConfigTable

This table supports the ARP Entry Configuration Table for 802.3 Media Cards and is a sequence of pdnNetToMediaConfigEntry where each entry is a list of information for each ARP Entry in the ARP Table to be displayed.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- pdnNetToMediaConfigIpAddr

OIDs:

- pdnNetToMediaConfigTable - { pdnNetToMediaConfig 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.1
- pdnNetToMediaConfigEntry - { pdnNetToMediaConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.1.1

Table 214. pdnNetToMediaConfigTable

Object	OID	Syntax	Access	Status	Supported
pdnNetToMediaConfigIpAddr	{ pdnNetToMediaConfigEntry 1 }	IpAddress	not-accessible	current	N
pdnNetToMediaConfigMacAddr	{ pdnNetToMediaConfigEntry 2 }	MacAddress	read-create	current	N
pdnNetToMediaConfigMin	{ pdnNetToMediaConfigEntry 3 }	INTEGER (0…99999)	read-only	current	N
pdnNetToMediaConfigFlags	{ pdnNetToMediaConfigEntry 4 }	Integer32	read-only	current	N
pdnNetToMediaConfigTrailer	{ pdnNetToMediaConfigEntry 5 }	SwitchState	read-create	current	N

Table 214. pdnNetToMediaConfigTable

Object	OID	Syntax	Access	Status	Supported
pdnNetToMediaConfigPerm	{ pdnNetToMediaConfigEntry 6 }	TruthValue	read-create	current	N
pdnNetToMediaConfigRowStatus	{ pdnNetToMediaConfigEntry 7 }	RowStatus	read-create	current	N

5.22.4.2.1.1 pdnNetToMediaConfigIpAddr

This objects corresponds to an IP Address of a Host or Client

5.22.4.2.1.2 pdnNetToMediaConfigMacAddr

This object corresponds to a MAC Address of a Host or Client

5.22.4.2.1.3 pdnNetToMediaConfigMin

This objects displays the number of minutes since an Arp entry was last used.

5.22.4.2.1.4 pdnNetToMediaConfigFlags

This object contains multiple information for an entry. The information can be interpreted as follows:

- 0x0004 - permanent entry
- 0x0008 - publish entry (respond for other host)
- 0x0010 - has requested trailers
- 0x0020 - Proxy Arp Entry
- 0x0040 - Subnetted Proxy ARP entry
- 0x0100 - Arp For Default Gateway

5.22.4.2.1.5 pdnNetToMediaConfigTrailer

This object is used to specify whether or not trailer encapsulation should be enabled or disabled.

5.22.4.2.1.6 pdnNetToMediaConfigPerm

This object is used to specify whether or not an entry should be stored permanently in Non-volatile storage. If the value is “true(1)” a permanent entry is added to non-volatile storage.

5.22.4.2.1.7 pdnNetToMediaConfigRowStatus

This object is used to create or delete a row in this table

5.22.4.2.2 pdnNetToMediaClearAllArp

Setting this object to a value of “clear(2)” will remove all entries from the ARP table.

5.22.4.2.3 pdnNetToMediaProxyArpTable

This table is used to enable or disable proxy ARP for the specified interface. This table is a sequence of pdnNetToMediaProxyArpEntry that allows a user to enable or disable proxy ARP for the specified interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- pdnNetToMediaProxyArpTable - { pdnNetToMediaConfig 3 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.3
- pdnNetToMediaProxyArpEntry - { pdnNetToMediaProxyArpTable 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.3.1

Table 215. pdnNetToMediaProxyArpTable

Object	OID	Syntax	Access	Status	Supported
pdnNetToMediaProxyArpStatus	{ pdnNetToMediaProxyArpEntry 1 }	INTEGER disable(1) enable(2)	read-create	current	N

5.22.4.2.4 pdnNetToMediaProxyArpStatus

Setting this object to a value “enable(2)” will enable proxy ARP for the interface identified by IfIndex. The value “disable(1)” will disable proxy ARP for the interface.

5.22.4.3 pdnNetTo8023MediaConfig

This group contains an ARP Entry Table for configuring 802.3 Media Cards

OIDs:

- pdnNetTo8023MediaConfig - { pdnNetToMediaGenericMIBObjects 3 } 1.3.6.1.4.1.1795.2.24.2.27.1.3

Table 216. pdnNetTo8023MediaConfig

Object	OID	Syntax	Access	Status	Supported
pdnNetTo8023MediaConfigTable	{ pdnNetTo8023MediaConfig 1 }	SEQUENCE of pdnNetTo8023MediaConfigEntry	not-accessible	current	N

5.22.4.3.1 pdnNetTo8023MediaConfigTable

A table displaying information about ARP Entries. This table is a sequence of pdnNetTo8032MediaConfigEntry where each entry is a list of information for each ARP Entry in the ARP Table to be displayed.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- pdnNetTo8023MediaConfigIpAddr
- pdnNetTo8023MediaConfigVnidId

OIDs:

- pdnNetTo8023MediaConfigTable - { pdnNetTo8023MediaConfig 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.3.1
- pdnNetTo8023MediaConfigEntry - { pdnNetTo8023MediaConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.3.1.1

Table 217. pdnNetTo8023MediaConfigTable

Object	OID	Syntax	Access	Status	Supported
pdnNetTo8023MediaConfigIpAddr	{ pdnNetTo8023MediaConfigEntry 1 }	IpAddress	not-accessible	current	N
pdnNetTo8023MediaConfigVnidId	{ pdnNetTo8023MediaConfigEntry 2 }	VnidRange	read-only	current	N
pdnNetTo8023MediaConfigMacAddr	{ pdnNetTo8023MediaConfigEntry 3 }	MacAddress	read-create	current	N
pdnNetTo8023MediaConfigMin	{ pdnNetTo8023MediaConfigEntry 4 }	INTEGER 1…99999)	read-only	current	N
pdnNetTo8023MediaConfigFlags	{ pdnNetTo8023MediaConfigEntry 5 }	Integer32	read-only	current	N

Table 217. pdnNetTo8023MediaConfigTable

Object	OID	Syntax	Access	Status	Supported
pdnNetTo8023MediaConfigTrailer	{ pdnNetTo8023MediaConfigEntry 6 }	SwitchState	read-create	current	N
pdnNetTo8023MediaConfigPerm	{ pdnNetTo8023MediaConfigEntry 7 }	TruthValue	read-create	current	N
pdnNetTo8023MediaConfigRowStatus	{ pdnNetTo8023MediaConfigEntry 8 }	RowStatus	read-create	current	N

5.22.4.3.1.1 pdnNetTo8023MediaConfigIpAddr

This objects corresponds to an IP Address of a Host or Client.

5.22.4.3.1.2 pdnNetTo8023MediaConfigVnidId

The VNID Id number of the virtual network for which this entry contains management information.

5.22.4.3.1.3 pdnNetTo8023MediaConfigMacAddr

This object corresponds to a MAC Address of a Host or Client.

5.22.4.3.1.4 pdnNetTo8023MediaConfigMin

This objects displays the number of minutes since an Arp entry was last used.

5.22.4.3.1.5 pdnNetTo8023MediaConfigFlags

This object contains multiple information for an entry. The information can be interpreted as follows:

- 0x0004 - permanent entry
- 0x0008 - publish entry (respond for other host)
- 0x0010 - has requested trailers
- 0x0020 - Proxy Arp Entry
- 0x0040 - Subnetted Proxy ARP entry
- 0x0100 - Arp For Default Gateway

5.22.4.3.1.6 pdnNetTo8023MediaConfigTrailer

This object is used to specify whether or not trailer encapsulation should be enabled or disabled.

5.22.4.3.1.7 pdnNetTo8023MediaConfigPerm

This object is used to specify whether or not an entry should be stored permanently in Non-volatile storage. If the value is “true(1)” a permanent entry is added to non-volatile storage.

5.22.4.3.1.8 pdnNetTo8023MediaConfigRowStatus

This object is used to create or delete a row in this table.

5.22.4.4 ipNetToMediaConfig

OIDs:

- ipNetToMediaConfig - { pdnNetToMediaConfig 4 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.4.4

Table 218. ipNetToMediaConfig

Object	OID	Syntax	Access	Status	Supported
ipNetToMediaForwardingMode	{ ipNetToMediaConfig 1 }	INTEGER basic(1) mxu(2) sms(3) ult(4)	read-write	current	Y
ipNetToMediaDefaultNHR	{ ipNetToMediaConfig 2 }	IpAddress	read-write	current	Y
ipNetToMediaExtTable	{ ipNetToMediaConfig 3 }	SEQUENCE OF ipNetToMediaExtEntry	not-accessible	current	Y
ipNetToMediaLimitTable	{ ipNetToMediaConfig 4 }	SEQUENCE OF ipNetToMediaLimitEntry	not-accessible	current	Y

5.22.4.4.1 ipNetToMediaForwardingMode

The following are values are defined for this object:

- basic(1) - basic or (transparent) Mode
- mxu(2) - Mux Forwarding Mode
  All traffic is routed to the upstream router. Port-to-port switching is not allowed.
  For more information about transparent vs. forwarding mode refer to RFC 1483 that defines bridge functionality and the mib RFC 1493 that defines mib needed to support different bridge functionality.
- sms(3) - Subscriber Management Mode
  In this mode all traffic is routed to the uplink port. Port-to-port switching is not allowed. This also enables other internal software functions that are needed to the SMS support. When this mode is selected, an external management VLAN is established to the Subscriber Management System.
- ult(4) - Uplink Tagging Mode
  Setting the value to ult(4) will enable Uplink Tagging. The Uplink Tagging Feature implements a portion of the IEEE 802.3Q VLAN standard to provide tagged Ethernet frames on the uplink port with a one-to-one mapping of ingress ADSL (subscriber) link to the VLAN identifier carried in the Ethernet tag header.

  In this mode, VLANs are assigned using the ultBaseVlanTag (Section 5.44.4.5, Page 505) and the ultIndex (Section 5.44.4.6, Page 505) MIB objects.

5.22.4.4.2 ipNetToMediaDefaultNHR

Setting this address will configure a default NHR for the system.

5.22.4.4.3 ipNetToMediaExtTable

This table is designed to augment ipNetToMediaTable from RFC 1213. This table is the IP Address Translation table used for mapping from IP addresses to physical addresses where each entry contains one IpAddress to ‘physical’ address equivalence.

Augments:

- ipNetToMediaTable (Section 5.4.3.1.22, Page 73)

OIDs:

- ipNetToMediaExtTable - { ipNetToMediaConfig 3 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.4.3
- ipNetToMediaExtEntry - { ipNetToMediaExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.4.3.1

Table 219. ipNetToMediaExtTable

Object	OID	Syntax	Access	Status	Supported
ipNetToMediaNHR	{ ipNetToMediaExtEntry 1 }	IpAddress	read-create	current	Y

5.22.4.4.3.1 ipNetToMediaNHR

This object is used to configure next hop router for the interface using proxy apr.

5.22.4.4.4 ipNetToMediaLimitTable

This table is used for to configure max number of IP addresses allowed on that interface. This table is a sequence of ipNetToMediaLimitEntry where each entry contains number of IP Addresses allowed on that interface.

Index:

- ipNetToMediaIfIndex (Section 5.4.3.1.22.1, Page 73)

OIDs:

- ipNetToMediaLimitTable - { ipNetToMediaConfig 4 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.4.4
- ipNetToMediaLimitEntry - { ipNetToMediaLimitTable 1 } 1.3.6.1.4.1.1795.2.24.2.27.1.2.4.4.1

Table 220. ipNetToMediaLimitTable

Object	OID	Syntax	Access	Status	Supported
ipNetToMediaLimitEnabled	{ ipNetToMediaLimitEntry 1 }	TruthValue	read-create	current	Y
ipNetToMediaMaxIPAddresses	{ ipNetToMediaLimitEntry 2 }	INTEGER(0…99999)	read-create	current	Y

5.22.4.4.4.1 ipNetToMediaLimitEnabled

This object is used to enable/disable limiting number of ip addresses on interfaces using proxy apr.

5.22.4.4.4.2 ipNetToMediaMaxIPAddresses

This object is used to configure number of ip addresses allowed on interfaces using proxy apr.

5.22.5 Traps

OIDs:

- pdnNetToMediaMIBTraps - { pdn-arp 2 } 1.3.6.1.4.1.1795.2.24.2.27.1

Table 221. PDN-ARP- MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
unauthorizedUserEvent	{ pdnNetToMediaMIBTraps 1 }	ipNetToMediaIfIndex ipNetToMediaPhysAddress	current	N

5.22.5.1 unauthorizedUserEvent

This trap is to communicate that on the interface where security authorization (MAC or IP to port) is enabled, frame arrived that is not authorized for that interface.

5.23 PDN-ATM-BRIDGE-IWF-MIB — PDN-ATM-BRIDGE-IWF-MIB.mib

5.23.1 Introduction

This MIB contains objects that are used for monitoring and controlling the ATM/Bridge Interworking Function.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200304240000Z,” or April 24, 2003.

5.23.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.23.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnAtmBridgeIwfMIB - { pdn-common 43 } 1.3.6.1.4.1.1795.2.24.2.43
- pdnAtmBridgeIwfNotifications - { pdnAtmBridgeIwfMIB 0 } 1.3.6.1.4.1.1795.2.24.2.43.0
- pdnAtmBridgeIwfObjects - { pdnAtmBridgeIwfMIB 1 } 1.3.6.1.4.1.1795.2.24.2.43

5.23.4 MIB Objects

Table 222. PDN-ATM-BRIDGE-IWF-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnAtmBridgeIwfTable	{ pdnAtmBridgeIwfObjects 1 }	SEQUENCE OF pdnAtmBridgeIwfEntry	not-accessible	current	Y

5.23.4.1 pdnAtmBridgeIwfTable

This table allows mapping of ATM VCs to Bridge Ports.

Indicies:

- ifIndex (Section 5.3.4.3.1, Page 37)
- pdnAtmBridgeIwfVclVpi

- pdnAtmBridgeIwfVclVci

OIDs:

- pdnAtmBridgeIwfTable - { pdnAtmBridgeIwfObjects 1 } 1.3.6.1.4.1.1795.2.24.2.43.1.1
- pdnAtmBridgeIwfEntry - { pdnAtmBridgeIwfTable 1 } 1.3.6.1.4.1.1795.2.24.2.43.1.1.1

Table 223. pdnAtmBridgeIwfTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmBridgeIwfVclVpi	{ pdnAtmBridgeIwfEntry 1 }	Unsigned32(0…4095)	not-accessible	current	Y
pdnAtmBridgeIwfVclVci	{ pdnAtmBridgeIwfEntry 2 }	Unsigned32(0…65535)	not-accessible	current	Y
pdnAtmBridgeIwfRowStatus	{ pdnAtmBridgeIwfEntry 3 }	RowStatus	read-create	current	Y
pdnAtmBridgeIwfDot1dBasePort	{ pdnAtmBridgeIwfEntry 4 }	Unsigned32(1…65535)	read-create	current	Y

5.23.4.1.1 pdnAtmBridgeIwfVclVpi

The VPI value of the VCL.

5.23.4.1.2 pdnAtmBridgeIwfVclVci

The VCI value of the VCL.

5.23.4.1.3 pdnAtmBridgeIwfRowStatus

The SNMP RowStatus (Section 4.6, Page 13) of the current row.

5.23.4.1.4 pdnAtmBridgeIwfDot1dBasePort

This object specifies the bridge port the VPI/VCI maps to.

5.23.5 Conformance Information

5.23.5.1 Compliance

No compliance statements were defined nor were any mandatory objects specified.

5.23.5.1.1 pdnAtmBridgeIwfMIBCompliance

The compliance statement for pdnAtmIwfBridge entities which implement the pdnAtmBridgeIwfMIB.

Mandatory Groups:

- pdnAtmBridgeIwfGroup

5.23.5.2 Conformance

Table 224. PDN-ATM-BRIDGE-IWF-MIB Conformance

Group	Description	Objects	Supported
pdnAtmBridgeIwfGroup	Objects grouped for ATM to Bridge Port Mapping	pdnAtmBridgeIwfRowStatus	Y
		pdnAtmBridgeIwfDot1dBasePort	Y

5.24 PDN-ATMEXT-MIB — pdn_AtmExt.mib

5.24.1 Introduction

This MIB Module extends the ATM-MIB defined in RFC 2515.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200303310000Z,” or March 31, 2003.

5.24.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.24.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2
- pdn-interfaces - { pdn-common 6 } 1.3.6.1.4.1.1795.2.24.2.6

- pdnAtm - { pdn-interfaces 11 } 1.3.6.1.4.1.1795.2.24.2.6.11
- pdnAtmExtMIB - { pdnAtm 5 } 1.3.6.1.4.1.1795.2.24.2.6.11.5
- pdnAtmExtMIBObjects - { pdnAtmExtMIB 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.1
- pdnAtmExtMIBTraps - { pdnAtmExtMIB 2 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.2
- pdnAtmExtMIBTrapPrefix - { pdnAtmExtMIBTraps 0 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.2.0
- pdnAtmExtMIBConformance - { pdnAtmExtMIB 3 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.3

5.24.4 MIB Objects

Table 225. PDN-ATMEXT-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnAtmIfConfExtTable	{ pdnAtmExtMIBObjects 1 }	SEQUENCE OF pdnAtmIfConfExtEntry	not-accessible	current	N
pdnAtmTrafficDescrParam ExtTable	{ pdnAtmExtMIBObjects 2 }	SEQUENCE OF pdnAtmTrafficDescrParamExtEntry	not-accessible	current	N
pdnAal5VccExtTable	{ pdnAtmExtMIBObjects 3 }	SEQUENCE OF pdnAal5VccExtEntry	not-accessible	current	Y

5.24.4.1 pdnAtmIfConfExtTable

The Paradyne Extension to the atmInterfaceConfTable of the ATM-MIB.

Augments:

- atmInterfaceConfTable (Section 5.13.4.1, Page 231)

Indicies:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- pdnAtmIfConfExtTable - { pdnAtmExtMIB 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.1
- pdnAtmIfConfExtEntry - { pdnAtmIfConfExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.1.1

Table 226. pdnAtmIfConfExtTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmIfConfExt VbrRtBandwidthUtil	{ pdnAtmIfConfExtEntry 1 }	Integer32(0…9999)	read-write	current	N
pdnAtmIfConfExt VbrNrtBandwidthUtil	{ pdnAtmIfConfExtEntry 2 }	Integer32(0…9999)	read-write	current	N
pdnAtmIfConfExt HecErrorThreshold	{ pdnAtmIfConfExtEntry 3 }	Integer32	read-write	current	N
pdnAtmIfConfExt UnknownCellThreshold	{ pdnAtmIfConfExtEntry 4 }	Integer32	read-write	current	N
pdnAtmIfConfExt OcdEventThreshold	{ pdnAtmIfConfExtEntry 5 }	Integer32	read-write	current	N
pdnAtmIfConfExt BandwidthUtilCbrReserved	{ pdnAtmIfConfExtEntry 6 }	Integer32	read-write	current	N
pdnAtmIfConfExt BandwidthUtilCbrAssigned	{ pdnAtmIfConfExtEntry 7 }	Integer32	read-only	current	N
pdnAtmIfConfExt BandwidthUtilVbrRtReserved	{ pdnAtmIfConfExtEntry 8 }	Integer32	read-write	current	N
pdnAtmIfConfExt BandwidthUtilVbrRtAssigned	{ pdnAtmIfConfExtEntry 9 }	Integer32	read-only	current	N
pdnAtmIfConfExt BandwidthUtilVbrNrtReserved	{ pdnAtmIfConfExtEntry 10 }	Integer32	read-write	current	N
pdnAtmIfConfExt BandwidthUtilVbrNrtAssigned	{ pdnAtmIfConfExtEntry 11 }	Integer32	read-only	current	N
pdnAtmIfConfExt BandwidthUtilUbrReserved	{ pdnAtmIfConfExtEntry 12 }	Integer32	read-write	current	N
pdnAtmIfConfExt BandwidthUtilUbrAssigned	{ pdnAtmIfConfExtEntry 13 }	Integer32	read-only	current	N
pdnAtmIfConfExt RateShape	{ pdnAtmIfConfExtEntry 14 }	Integer32(512…46080)	read-write	current	N

5.24.4.1.1 pdnAtmIfConfExtVbrRtBandwidthUtil

The bandwidth utilization for real-time vbr type traffic expressed as a percentage.

5.24.4.1.2 pdnAtmIfConfExtVbrNrtBandwidthUtil

The bandwidth utilization for non-real-time VBR type traffic expressed as a percentage.

5.24.4.1.3 pdnAtmIfConfExtHecErrorThreshold

The value of this object represents the threshold value for the number of uncorrectable HEC errors in a 15 minutes interval. An SNMP link Up/Down trap would be generated if the number of uncorrectable HEC errors in a 15 minutes interval exceeds this threshold. The valid range of this object is 0 to 1000. A value of 0 will cause a trap to be sent on each occurrence of a HEC error. A value of -9999 disables the threshold check.

The MIB recommended default value for this object is 100.

5.24.4.1.4 pdnAtmIfConfExtUnknownCellThreshold

The value of this object represents the threshold value for the number of unknown cells in a 15 minutes interval. An SNMP trap would be generated if the number of unknown cell in a 15 minutes interval exceeds this threshold. The valid range of this object is 0 to 1000. A value of 0 will cause a trap to be sent on each occurrence of an unknown cell. A value of -9999 disables the threshold check.

The MIB recommended default value for this object is 10.

5.24.4.1.5 pdnAtmIfConfExtOcdEventThreshold

The value of this object represents the threshold value for the number of Out of Cell Delineation(OCD) events in a 15 minutes interval. An SNMP trap would be generated if the number of OCD events in a 15 minutes interval exceeds this threshold. The valid range of this object is 0 to 1000. A value of 0 will cause a trap to be sent on each occurrence of a HEC error. A value of -9999 disables the threshold check.

The MIB recommended default value for this object is 0.

5.24.4.1.6 pdnAtmIfConfExtBandwidthUtilCbrReserved

This object specifies the percent of the interface bandwidth to be allocated to the CBR traffic class.

5.24.4.1.7 pdnAtmIfConfExtBandwidthUtilCbrAssigned

This object specifies the percent of bandwidth utilization (sometimes referred to as over subscription) the connection Admission Control algorithm has already allowed for the CBR class of service.

5.24.4.1.8 pdnAtmIfConfExtBandwidthUtilVbrRtReserved

This object specifies the percent of the interface bandwidth to be allocated to the VBR-rt traffic class.

5.24.4.1.9 pdnAtmIfConfExtBandwidthUtilVbrRtAssigned

This object specifies the percent of bandwidth utilization (sometimes referred to as over subscription) the connection Admission Control algorithm has already allowed for the VBR-rt class of service.

5.24.4.1.10 pdnAtmIfConfExtBandwidthUtilVbrNrtReserved

This object specifies the percent of the interface bandwidth to be allocated to the VBR-nrt traffic class.

5.24.4.1.11 pdnAtmIfConfExtBandwidthUtilVbrNrtAssigned

This object specifies the percent of bandwidth utilization (sometimes referred to as over subscription) the connection Admission Control algorithm has already allowed for the VBR-nrt class of service.

5.24.4.1.12 pdnAtmIfConfExtBandwidthUtilUbrReserved

This object specifies the percent of the interface bandwidth to be allocated to the UBR traffic class.

5.24.4.1.13 pdnAtmIfConfExtBandwidthUtilUbrAssigned

This object specifies the percent of bandwidth utilization (sometimes referred to as over subscription) the connection Admission Control algorithm has already allowed for the UBR class of service.

5.24.4.1.14 pdnAtmIfConfExtRateShape

This object is used to specify rate shaping on this interface in units of kbps. The valid range of this object is 512 to 46080 [512K…46Mbps] with increments of 64K. The default value is set depending on the interface.

5.24.4.2 pdnAtmTrafficDescrParamExtTable

The Paradyne Extension to the atmTrafficDescrParamTable of the ATM-MIB.

Augments:

- atmTrafficDescrParamTable (Section 5.13.4.4, Page 235)

Indicies:

- atmTrafficDescrParamIndex (Section 5.13.4.4.1, Page 236)

OIDs:

- pdnAtmTrafficDescrParamExtTable - { pdnAtmExtMIB 2 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.2
- pdnAtmTrafficDescrParamExtEntry - { pdnAtmTrafficDescrParamExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.2.1

Table 227. pdnAtmTrafficDescrParamExtTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmTrafficDescrParamName	{ pdnAtmTrafficDescrParamExtEntry 1 }	DisplayString (SIZE(1…12))	read-create	current	N
pdnAtmTrafficPolicing	{ pdnAtmTrafficDescrParamExtEntry 2 }	TruthValue	read-create	current	N

5.24.4.2.1 pdnAtmTrafficDescrParamName

The value of this Object identifies the traffic descriptor by a by a name.

5.24.4.2.2 pdnAtmTrafficPolicing

The value of this Object determines whether or not the vcl/vpl to which this traffic descriptor is bound will be policed.

5.24.4.3 pdnAal5VccExtTable

The Paradyne Extension to the aal5VccTable in the ATM-MIB.

Augments:

- aal5VccTable (Section 5.13.4.11, Page 251)

Indicies:

- ifIndex (Section 5.3.4.3.1, Page 37)
- aal5VccVpi
- aal5VccVci
-

OIDs:

- pdnAal5VccExtTable - { pdnAtmExtMIB 3 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.3
- pdnAal5VccExtEntry - { pdnAal5VccExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.3.1

Table 228. pdnAal5VccExtTable

Object	OID	Syntax	Access	Status	Supported
pdnAal5VccExtOutPDUs	{ pdnAal5VccExtEntry 1 }	Unsigned32	read-only	current	Y
pdnAal5VccExtInPDUs	{ pdnAal5VccExtEntry 2 }	Unsigned32	read-only	current	Y

5.24.4.3.1 pdnAal5VccExtOutPDUs

The total number of AAL5 CPCS PDUs transmitted on the interface associated with an AAL5 entry.

5.24.4.3.2 pdnAal5VccExtInPDUs

This total number of AAL5 CPCS PDUs transmitted on the interface associated with an AAL5 entry.

5.24.5 Traps

This section defines the traps generated.

OIDs:

- pdnAtmExtMIBTraps - { pdnAtmExtMIB 2 } 1.3.6.1.4.1.1795.2.24.2.6.11.5.2

Table 229. ADSL-ATMEXT-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
pdnAtmIfConfExt ExcessInvalidCellsAlarm	{ pdnAtmExtMIBTrapPrefix 1 }	ifOperStatus pdnAtmIfConfExtUnknownCellThreshold	current	N
pdnAtmIfConfExt EgressLineRateAlarmSet	{ pdnAtmExtMIBTrapPrefix 2 }	ifIndex	current	N
pdnAtmIfConfExt EgressLineRateAlarmClear	{ pdnAtmExtMIBTrapPrefix 102 }	ifIndex	current	N
pdnAtmIfConfExt VplNoBandwidthAvail	{ pdnAtmExtMIBTrapPrefix 3 }	atmVpCrossConnectAdminStatus	current	N
pdnAtmIfConfExt VclNoBandwidthAvail	{ pdnAtmExtMIBTrapPrefix 4 }	atmVcCrossConnectAdminStatus	current	N
pdnAtmIfConfExt IngressLineRateAlarmSet	{ pdnAtmExtMIBTrapPrefix 5 }	ifIndex	current	N
pdnAtmIfConfExt IngressLineRateAlarmClear	{ pdnAtmExtMIBTrapPrefix 105 }	ifIndex	current	N

5.24.5.1 pdnAtmIfConfExtExcessInvalidCellsAlarm

This alarm indicates that the number of cells with uncorrectable HEC errors has exceeded the used defined the threshold.

5.24.5.2 pdnAtmIfConfExtEgressLineRateAlarmSet

This alarm indicates that the line rate of the specified interface has dropped below a value which violates the traffic contracts of the admitted connections in the egress (transmit) direction.

5.24.5.3 pdnAtmIfConfExtEgressLineRateAlarmClear

This alarm indicates that the line rate of the specified interface has risen above a value which would no longer violate the traffic contracts of the admitted connections in the egress (transmit) direction.

5.24.5.4 pdnAtmIfConfExtVplNoBandwidthAvail

This alarm indicates that the traffic contract(s) associated with the Virtual Path (VP) Cross Connect cannot be satisfied by the ATM switch. It is left up to the implementation to decide whether or not the VP cross connect will be admitted.

5.24.5.5 pdnAtmIfConfExtVclNoBandwidthAvail

This alarm indicates that the traffic contract(s) associated with the Virtual Channel (VC) Cross Connect cannot be satisfied by the ATM switch. It is left up to the implementation to decide whether or not the VC cross connect will be admitted.

5.24.5.6 pdnAtmIfConfExtIngressLineRateAlarmSet

This alarm indicates that the line rate of the specified interface has dropped below a value which violates the traffic contracts of the admitted connections in the ingress (receive) direction.

5.24.5.7 pdnAtmIfConfExtIngressLineRateAlarmClear

This alarm indicates that the line rate of the specified interface has risen above a value which would no longer violate the traffic contracts of the admitted connections in the ingress (receive) direction.

5.24.6 Conformance Information

5.24.7 Compliance

No compliance statements were defined nor were any mandatory objects specified.

5.24.8 Conformance

Table 230. ADSL-ATMEXT-MIB Conformance

Group	Description	Objects	Supported
pdnAtmIfConfExtGroup	Objects grouped for the pdnAtmIfConfExtTable.	pdnAtmIfConfExtVbrRtBandwidthUtil	N
		pdnAtmIfConfExtVbrNrtBandwidthUtil	N
		pdnAtmIfConfExtHecErrorThreshold	N
		pdnAtmIfConfExtUnknownCellThreshold	N
		pdnAtmIfConfExtOcdEventThreshold	N
		pdnAtmIfConfExtBandwidthUtilCbrReserved	N
		pdnAtmIfConfExtBandwidthUtilCbrAssigned	N
		pdnAtmIfConfExtBandwidthUtilVbrRtReserved	N
		pdnAtmIfConfExtBandwidthUtilVbrRtAssigned	N
		pdnAtmIfConfExtBandwidthUtilVbrNrtReserved	N
		pdnAtmIfConfExtBandwidthUtilVbrNrtAssigned	N
		pdnAtmIfConfExtBandwidthUtilUbrReserved	N
		pdnAtmIfConfExtBandwidthUtilUbrAssigned	N
		pdnAtmIfConfExtRateShape	N
pdnAtmTrafficDescrParamsExtGroup	Objects grouped for the pdnAtmTrafficDescrParamExtTable	pdnAtmTrafficDescrParamName	N
		pdnAtmTrafficPolicing	Y
pdnAal5VccExtGroup	Objects grouped for the pdnAal5VccExtTable.	pdnAal5VccExtInPDUs	Y
		pdnAal5VccExtOutPDUs	N
pdnAtmExtNotificationGroup	Objects grouped for Notification defined in this MIB.	pdnAtmIfConfExtExcessInvalidCellsAlarm	N
		pdnAtmIfConfExtEgressLineRateAlarmSet	N
		pdnAtmIfConfExtEgressLineRateAlarmClear	N
		pdnAtmIfConfExtVplNoBandwidthAvail	N
		pdnAtmIfConfExtVclNoBandwidthAvail	N
		pdnAtmIfConfExtIngressLineRateAlarmSet	N
		pdnAtmIfConfExtIngressLineRateAlarmClear	N

5.25 PDN-ATMSTATS-MIB — pdn_AtmStats.mib

5.25.1 Introduction

This MIB Module supports ATM Statistics.

The version of the MIB used for this document had a LAST-UPDATED of 0004130000Z which decodes to April 13, 2000.

5.25.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.25.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-interfaces - { pdn-common 6 } 1.3.6.1.4.1.1795.2.24.2.6
- pdnAtm - { pdn-interfaces 11 } 1.3.6.1.4.1.1795.2.24.2.6.11
- pdnAtmStatsMIB - { pdnAtm 3 } 1.3.6.1.4.1.1795.2.24.2.6.11.3
- pdnAtmVplStat - { pdnAtmStatsMIB 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.1
- pdnAtmVclStat - { pdnAtmStatsMIB 2 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2
- pdnAtmStat - { pdnAtmStatsMIB 3 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.3

5.25.4 MIB Objects

Table 231. PDN-ATMSTATS-MIB Objects

Object	OID	Description	Supported
pdnAtmVplStat	{ pdnAtmStatsMIB 1 }	ATM VPL Statistics	N
pdnAtmVclStat	{ pdnAtmStatsMIB 2 }	ATM VCL Statistics	Y
pdnAtmStat	{ pdnAtmStatsMIB 3 }	ATM Statistics	Y

5.25.4.1 pdnAtmVplStat

This group consists of VPL statistics.

OIDs:

- pdnAtmVplStat - { pdnAtmStatsMIB 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.1

Table 232. pdnAtmVplStat

Object	OID	Syntax	Access	Status	Supported
pdnAtmVplStatTable	{ pdnAtmVplStat 1 }	SEQUENCE OF pdnAtmVplStatEntry	not-accessible	current	N

5.25.4.1.1 pdnAtmVplStatTable

This table contains all statistic counters per VPL. It is used to monitor the usage of the VPL in terms of incoming cells and outgoing cells. This table is a sequence of pdnAtmVlpStatEntry where each entry in this table represents a VPL.

Indices:

- pdnAtmVplStatIfIndex
- pdnAtmVplStatVpi

OIDs:

- pdnAtmVplStatTable - { pdnAtmVplStat 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.1.1
- pdnAtmVplStatEntry - { pdnAtmVplStatTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.1.1.1

Table 233. pdnAtmVplStatTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmVplStatIfIndex	{ pdnAtmVplStatEntry 1 }	Integer32	read-only	current	N
pdnAtmVplStatVpi	{ pdnAtmVplStatEntry 2 }	Integer32	read-only	current	N
pdnAtmVplStatTotalCellIns	{ pdnAtmVplStatEntry 3 }	Counter32	read-only	current	N
pdnAtmVplStatClp0CellIns	{ pdnAtmVplStatEntry 4 }	Counter32	read-only	current	N
pdnAtmVplStatTotalDiscards	{ pdnAtmVplStatEntry 5 }	Counter32	read-only	current	N
pdnAtmVplStatClp0Discards	{ pdnAtmVplStatEntry 6 }	Counter32	read-only	current	N
pdnAtmVplStatTotalCellOuts	{ pdnAtmVplStatEntry 7 }	Counter32	read-only	current	N
pdnAtmVplStatClp0CellOuts	{ pdnAtmVplStatEntry 8 }	Counter32	read-only	current	N
pdnAtmVplStatTaggedOuts	{ pdnAtmVplStatEntry 9 }	Counter32	read-only	current	N

5.25.4.1.1.1 pdnAtmVplStatIfIndex

A unique value for each interface. Its value ranges 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 reinitialization.

5.25.4.1.1.2 pdnAtmVplStatVpi

The VPI value of the VPL.

5.25.4.1.1.3 pdnAtmVplStatTotalCellIns

The total number of valid ATM cells received by this VPL including both CLP=0 and CLP=1 cells. The cells are counted prior to the application of the traffic policing.

5.25.4.1.1.4 pdnAtmVplStatClp0CellIns

The number of valid ATM cells received by this VPL with CLP=0. The cells are counted prior to the application of the traffic policing.

5.25.4.1.1.5 pdnAtmVplStatTotalDiscards

The total number of valid ATM cells discarded by the traffic policing entity. This includes cells originally received with CLP=0 and CLP=1.

5.25.4.1.1.6 pdnAtmVplStatClp0Discards

The total number of valid ATM cells received with CLP=0 an discarded by the traffic policing entity.

5.25.4.1.1.7 pdnAtmVplStatTotalCellOuts

The total number of valid ATM cells transmitted by this VPL. This includes both CLP=0 and CLP=1 cells.

5.25.4.1.1.8 pdnAtmVplStatClp0CellOuts

The total number of valid ATM cells transmitted with CLP=0 by this VPL.

5.25.4.1.1.9 pdnAtmVplStatTaggedOuts

The total number of valid ATM cells tagged by the traffic policing entity from CLP=0 to CLP=1 and transmitted by this VPL.

5.25.4.2 pdnAtmVclStat

This group consists of Vcl statistics.

OIDs:

- pdnAtmVclStat - { pdnAtmStatsMIB 2 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2

Table 234. pdnAtmVclStat

Object	OID	Syntax	Access	Status	Supported
pdnAtmVclStatTable	{ pdnAtmVclStat 2 }	SEQUENCE OF pdnAtmVclStatEntry	not-accessible	current	N
pdnAtmVclCurrTable	{ pdnAtmVclStat 3 }	SEQUENCE OF pdnAtmVclCurrEntry	not-accessible	current	N
pdnAtmVclHistTable	{ pdnAtmVclStat 4 }	SEQUENCE OF pdnAtmVclHistEntry	not-accessible	current	N

5.25.4.2.1 pdnAtmVclStatTable

This table contains all statistic counters per VCL. It is used to monitor the usage of the VCL in terms of incoming cells and outgoing cells. The table is a sequence of pdnAtmVclStatEntry where each entry in this table represents a VCL.

Indices:

- pdnAtmVclStatIfIndex
- pdnAtmVclStatVpi
- pdnAtmVclStatVci

OIDs:

- pdnAtmVclStatTable - { pdnAtmVclStat 2 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2.2
- pdnAtmVclStatEntry - { pdnAtmVclStatTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2.2.1

Table 235. pdnAtmVclStatTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmVclStatIfIndex	{ pdnAtmVclStatEntry 1 }	Integer32	read-only	current	N
pdnAtmVclStatVpi	{ pdnAtmVclStatEntry 2 }	Integer32	read-only	current	N
pdnAtmVclStatVci	{ pdnAtmVclStatEntry 3 }	Integer32	read-only	current	N
pdnAtmVclStatTotalCellIns	{ pdnAtmVclStatEntry 4 }	Counter32	read-only	current	N
pdnAtmVclStatClp0CellIns	{ pdnAtmVclStatEntry 5 }	Counter32	read-only	current	N
pdnAtmVclStatTotalDiscards	{ pdnAtmVplStatEntry 6 }	Counter32	read-only	current	N
pdnAtmVclStatClp0Discards	{ pdnAtmVclStatEntry 7 }	Counter32	read-only	current	N
pdnAtmVclStatTotalCellOuts	{ pdnAtmVclStatEntry 8 }	Counter32	read-only	current	N
pdnAtmVclStatClp0CellOuts	{ pdnAtmVclStatEntry 9 }	Counter32	read-only	current	N
pdnAtmVclStatTaggedOuts	{ pdnAtmVclStatEntry 10 }	Counter32	read-only	current	N
pdnAtmVclStatRowStatus	{ pdnAtmVclStatEntry 11 }	RowStatus	read-write	current	N

5.25.4.2.1.1 pdnAtmVclStatIfIndex

A unique value for each interface. Its value ranges 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.

5.25.4.2.1.2 pdnAtmVclStatVpi

The VPI value of the VCL.

5.25.4.2.1.3 pdnAtmVclStatVci

The VCI value of the VCL.

5.25.4.2.1.4 pdnAtmVclStatTotalCellIns

The total number of valid ATM cells received by this VCL including both CLP=0 and CLP=1 cells. The cells are counted prior to the application of the traffic policing.

5.25.4.2.1.5 pdnAtmVclStatClp0CellIns

The number of valid ATM cells received by this VCL with CLP=0. The cells are counted prior to the application of the traffic policing.

5.25.4.2.1.6 pdnAtmVclStatTotalDiscards

The total number of valid ATM cells discarded by the traffic policing entity. This includes cells originally received with CLP=0 and CLP=1.

5.25.4.2.1.7 pdnAtmVclStatClp0Discards

The total number of valid ATM cells received with CLP=0 an discarded by the traffic policing entity.

5.25.4.2.1.8 pdnAtmVclStatTotalCellOuts

The total number of valid ATM cells transmitted by this VCL. This includes both CLP=0 and CLP=1 cells.

5.25.4.2.1.9 pdnAtmVclStatClp0CellOuts

The total number of valid ATM cells transmitted with CLP=0 by this VCL.

5.25.4.2.1.10 pdnAtmVclStatTaggedOuts

The total number of valid ATM cells tagged by the traffic policing entity from CLP=0 to CLP=1 and transmitted by this VCL.

5.25.4.2.1.11 pdnAtmVclStatRowStatus

This object is used to create a new row or delete an existing row in this table. When a new row is added, statistics are collected for the VCL. Statistics collection is terminated by deletion of the row.

5.25.4.2.2 pdnAtmVclCurrTable

This table maintains per-TP statistics for the fifteen-minute interval currently being collected. This table is a sequence of pdnAtmVclCurrEntry where an entry in this table for VCL termination points. Each VCL Termination Point automatically has an entry in this table associated with it.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- atmVclVpi (Section 5.13.4.6.1, Page 242)
- atmVclVci (Section 5.13.4.6.2, Page 242)

OIDs:

- pdnAtmVclCurrTable - { pdnAtmVclStat 3 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2.3
- pdnAtmVclCurrEntry - { pdnAtmVclCurrTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2.3.1

Table 236. pdnAtmVclCurrTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmVclCurrElapsedTime	{ pdnAtmVclCurrEntry 1 }	TimeInterval	read-only	current	N
pdnAtmVclCurrTotalCellIns	{ pdnAtmVclCurrEntry 2 }	Gauge32	read-only	current	N
pdnAtmVclCurrTotalInDiscards	{ pdnAtmVclCurrEntry 3 }	Gauge32	read-only	current	N
pdnAtmVclCurrTotalCellOuts	{ pdnAtmVclCurrEntry 4 }	Gauge32	read-only	current	N
pdnAtmVclCurrTotalOutDiscards	{ pdnAtmVclCurrEntry 5 }	Gauge32	read-only	current	N

5.25.4.2.2.1 pdnAtmVclCurrElapsedTime

Amount of time, measured in seconds, that statistics for this entry (the current interval) have been counted.

5.25.4.2.2.2 pdnAtmVclCurrTotalCellIns

The total number of valid ATM cells received by this VCL. This number corresponds to that of total cells received from CPE if this interface is a DSL ATM interface.

5.25.4.2.2.3 pdnAtmVclCurrTotalInDiscards

The number of discarded received cells.

5.25.4.2.2.4 pdnAtmVclCurrTotalCellOuts

The number of cells that have been successfully sent. This number corresponds to that of total cells transmitted to CPE if this interface is a DSL ATM interface.

5.25.4.2.2.5 pdnAtmVclCurrTotalOutDiscards

The number of discarded transmitted cells since this interval.

5.25.4.2.3 pdnAtmVclHistTable

This table maintains per-interface statistics for previous fifteen-minute intervals. The table is a sequence of pdnAtmVclHistory where an entry in History Data table for VCL termination points. Each VCL Termination Point automatically has an entry in this table associated with it. This data are only recorded fro permanent connections.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- atmVclVpi (Section 5.13.4.6.1, Page 242)
- atmVclVci (Section 5.13.4.6.2, Page 242)
- pdnAtmVclHistIndex

OIDs:

- pdnAtmVclHistTable - { pdnAtmVclStat 4 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2.4
- pdnAtmVclHistEntry - { pdnAtmVclHistTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.2.4.1

Table 237. pdnAtmVclHistTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmVclHistIndex	{ pdnAtmVclHistEntry 1 }	Integer32(1…96)	not-accessible	current	N
pdnAtmVclHistElapsedTime	{ pdnAtmVclHistEntry 2 }	TimeInterval	read-only	current	N
pdnAtmVclHistTotalCellIns	{ pdnAtmVclHistEntry 3 }	Gauge32	read-only	current	N
pdnAtmVclHistTotalInDiscards	{ pdnAtmVclHistEntry 4 }	Gauge32	read-only	current	N
pdnAtmVclHistTotalCellOuts	{ pdnAtmVclHistEntry 5 }	Gauge32	read-only	current	N
pdnAtmVclHistTotalOutDiscards	{ pdnAtmVclHistEntry 6 }	Gauge32	read-only	current	N

5.25.4.2.3.1 pdnAtmVclHistIndex

A number between 1 and 96, which identifies the interval for which the set of statistics in this entry was collected. The interval identified by 1 is the most recently completed 15 minute interval, and the interval identified by N is the interval immediately preceding the one identified by N-1.

5.25.4.2.3.2 pdnAtmVclHistElapsedTime

Amount of Time, measured in units of 0.01 second, that statistics for this entry (the current interval) have been counted.

5.25.4.2.3.3 pdnAtmVclHistTotalCellIns

The total number of valid ATM cells received by this VCL during this interval. This number corresponds to that of total cells received from CPE if this interface is a DSL ATM interface.

5.25.4.2.3.4 pdnAtmVclHistTotalInDiscards

The number of discarded received cells during this interval.

5.25.4.2.3.5 pdnAtmVclHistTotalCellOuts

The number of cells that have been successfully sent during this interval. This number corresponds to that of total cells transmitted to CPE if this interface is a DSL ATM interface.

5.25.4.2.3.6 pdnAtmVclHistTotalOutDiscards

The number of discarded transmitted cells during this interval.

5.25.4.3 pdnAtmStat

Atm Statistics group.

OIDs:

- pdnAtmStat - { pdnAtmStatsMIB 3 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.3

Table 238. pdnAtmStat

Object	OID	Syntax	Access	Status	Supported
pdnAtmStatTable	{ pdnAtmVclStat 2 }	SEQUENCE OF pdnAtmStatEntry	not-accessible	current	Y

5.25.4.3.1 pdnAtmStatTable

This table contains all statistic counters per interface. It is used to monitor the errors of the in terms of incoming cells. The table is a sequence of pdnAtmStatEntry where each entry in this table represents a error count per interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- pdnAtmStatTable - { pdnAtmStat 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.3.1
- pdnAtmStatEntry - { pdnAtmStatTable 1 } 1.3.6.1.4.1.1795.2.24.2.6.11.3.3.1.1

Table 239. pdnAtmStatTable

Object	OID	Syntax	Access	Status	Supported
pdnAtmStatHECErrors	{ pdnAtmStatEntry 1 }	Counter32	read-only	current	Y
pdnAtmStatLCDErrors	{ pdnAtmStatEntry 2 }	Counter32	read-only	current	Y

5.25.4.3.1.1 pdnAtmStatHECErrors

The total number of ATM cells discarded on this interface due to a HEC violation.

5.25.4.3.1.2 pdnAtmStatLCDErrors

The total number of Loss of Cell Delimitation events (LCD) that occurred on this interface.

5.26 PDN-CONFIG-MIB — pdn_Config.mib

5.26.1 Introduction

This MIB Module contains general objects for configuring devices.

The version of the MIB used for this document had a comment that stated its version date as December 13, 2001.

5.26.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- DateAndTime - A date-time specification. The display hint is “2d-1d-1d,1d:1d:1d.1d,1a1d:1d.”

Field	Octets	Contends	Range
1	1-2	year[a]	0…65536
2	3	month	1…12
3	4	day	1…31
4	5	hour	0…23
5	6	minutes	0…59
6	7	seconds	0…60[b]
7	8	deci-seconds	0…9
8	9	direction from UTC	“+” / “-”
9	10	hours from UTC[c]	0…13
10	11	minutes from UTC	0…59

a. The value if year is in network-byte order

b. Use 60 for leap-second

c. Daylight saving time in New Zealand is +13

For example, Tuesday May 26, 1992 at 1:30:15 PM EDT would be displayed as: 1992-5-26,13:30:15.0,-4:0.
Note that if only local time is known, then timezone information (fields 8-10) is not present.

The syntax for this object is:

— OCTET STRING (SIZE (8 | 11))

5.26.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24

- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-devConfig - { pdn-common 7 } 1.3.6.1.4.1.1795.2.24.2.7
- devConfigArea - { pdn-devConfig 1 } 1.3.6.1.4.1.1795.2.24.2.7.1
- devConfigTestTimer - { pdn-devConfig 2 } 1.3.6.1.4.1.1795.2.24.2.7.2
- devConfigClockSrc - { pdn-devConfig 3 } 1.3.6.1.4.1.1795.2.24.2.7.3
- devConfigTrap - { pdn-devConfig 4 } 1.3.6.1.4.1.1795.2.24.2.7.4
- devConfigAlarm - { pdn-devConfig 5 } 1.3.6.1.4.1.1795.2.24.2.7.5
- devConfigCardType - { pdn-devConfig 6 } 1.3.6.1.4.1.1795.2.24.2.7.6
- devConfigNetSync - { pdn-devConfig 7 } 1.3.6.1.4.1.1795.2.24.2.7.7
- devConfigTime - { pdn-devConfig 8 } 1.3.6.1.4.1.1795.2.24.2.7.8
- devConfigChangeKeys - { pdn-devConfig 9 } 1.3.6.1.4.1.1795.2.24.2.7.9
- devConfiguration - { pdn-devConfig 10 } 1.3.6.1.4.1.1795.2.24.2.7.10

5.26.4 MIB Objects

Table 240. PDN-CONFIG-MIB Objects

Object	OID	Description	Supported
devConfigArea	{ pdn-devConfig 1 }	The device Configuration Area Group	N
devConfigTestTimer	{ pdn-devConfig 2 }	The Test Timeout Group.	Y
devConfigClockSrc	{ pdn-devConfig 3 }	The Clock Source Group	N
devConfigTrap	{ pdn-devConfig 4 }	The Trap Configuration Group	N
devConfigAlarm	{ pdn-devConfig 5 }	The System Alarm Group	N
devConfigCardType	{ pdn-devConfig 6 }	The Card Type Table	N
devConfigNetSync	{ pdn-devConfig 7 }	The Device Network Synchronization Group	N
devConfigTime	{ pdn-devConfig 8 }	The Device Configuration Time Group	Y
devConfigChangeKeys	{ pdn-devConfig 9 }	The Device Configuration CHange Key Group	N
devConfiguration	{ pdn-devConfig 10 }	The General Configuration Group	Y

5.26.4.1 devConfigArea

OIDs:

- devConfigArea - { pdn-devConfig 1 } 1.3.6.1.4.1.1795.2.24.2.7.1

Table 241. devConfigArea

Object	OID	Syntax	Access	Status	Supported
devConfigAreaCopy	{ pdn-devConfig 1 }	INTEGER noOp(1) active-to-customer1(2) active-to-customer2(3) customer1-to-active(4) customer1-to-customer2(5) customer2-to-active(6) customer2-to-customer1(7) factory1-to-active(8) factory1-to-customer1(9) factory1-to-customer2(10) factory2-to-active(11) factory2-to-customer1(12) factory2-to-customer2(13) factory3-to-active(14) factory3-to-customer1(15) factory3-to-customer2(16) factory4-to-active(17) factory4-to-customer1(18) factory4-to-customer2(19) factory5-to-active(20) factory5-to-customer1(21) factory5-to-customer2(22) factory6-to-active(23) factory6-to-customer1(24) factory6-to-customer2(25) factory7-to-active(26) factory7-to-customer1(27) factory7-to-customer2(28) factory8-to-active(29) factory8-to-customer1(30) factory8-to-customer2(31) factory9-to-active(32) factory9-to-customer1(33) factory9-to-customer2(34)	read-write	mandatory	N

5.26.4.1.1 devConfigAreaCopy

This object is used to copy the entire contents of one configuration area into another configuration area. The supported number of factory configuration areas depends on the device model. The value read from this object is always noOp(1). The following values are defined for this object:

- noOp(1)
- active-to-customer1(2)
- active-to-customer2(3)
- customer1-to-active(4)
- customer1-to-customer2(5)
- customer2-to-active(6)
- customer2-to-customer1(7)

- factory1-to-active(8)
- factory1-to-customer1(9)
- factory1-to-customer2(10)
- factory2-to-active(11)
- factory2-to-customer1(12)
- factory2-to-customer2(13)
- factory3-to-active(14)
- factory3-to-customer1(15)
- factory3-to-customer2(16)
- factory4-to-active(17)
- factory4-to-customer1(18)
- factory4-to-customer2(19)
- factory5-to-active(20)
- factory5-to-customer1(21)
- factory5-to-customer2(22)
- factory6-to-active(23)
- factory6-to-customer1(24)
- factory6-to-customer2(25)
- factory7-to-active(26)
- factory7-to-customer1(27)
- factory7-to-customer2(28)
- factory8-to-active(29)
- factory8-to-customer1(30)
- factory8-to-customer2(31)
- factory9-to-active(32)
- factory9-to-customer1(33)
- factory9-to-customer2(34)

5.26.4.2 devConfigTestTimer

The Test Timeout group. This group is used for configuring timers associated with tests on the device.

OIDs:

- devConfigTestTimer - { pdn-devConfig 2 } 1.3.6.1.4.1.1795.2.24.2.7.2

Table 242. devConfigTestTimer

Object	OID	Syntax	Access	Status	Supported
devConfigTestTimeout	{ devConfigTestTimer 1 }	INTEGER disable(1) enable(2)	read-write	mandatory	Y
devConfigTestDuration	{ devConfigTestTimer 2 }	INTEGER	read-write	mandatory	Y

5.26.4.2.1 devConfigTestTimeout

The following values are defined for this object:

- disable(1) - Tests will not be terminated based on a timer.
- enable(2) - Tests are terminated after the duration specified by devConfigTestDuration

This object specifies whether tests are to be terminated after a defined duration. If the value is enable(2), the duration is defined by devConfigTestDuration.

5.26.4.2.2 devConfigTestDuration

This object specifies the duration that a test is allowed to run before it is automatically terminated. Tests will only be terminated if devConfigTestTimeout is set to enable.

5.26.4.3 devConfigClockSrc

The Clock Source group.

OIDs:

- devConfigClockSrc - { pdn-devConfig 3 } 1.3.6.1.4.1.1795.2.24.2.7.3

Table 243. devConfigClockSrc

Object	OID	Syntax	Access	Status	Supported
devConfigClockSrcTable	{ devConfigClockSrc 1 }	SEQUENCE OF devConfigClockSrcEntry	not-accessible	mandatory	N

5.26.4.3.1 devConfigClockSrcTable

The clock source table. This table is a sequence of devConfigClockSrcEntry where each entry contains pertinent data on configuring clock sources.

Index:

- devCfgClkWhichSrc

OIDs:

- devConfigClockSrcTable - { devConfigClockSrc 1 } 1.3.6.1.4.1.1795.2.24.2.7.3.1
- devConfigClockSrcEntry - { devConfigClockSrcTable 1 } 1.3.6.1.4.1.1795.2.24.2.7.3.1.1

Table 244. devConfigClockSrcTable

Object	OID	Syntax	Access	Status	Supported
devCfgClkWhichSrc	{ devConfigClockSrcEntry 1 }	INTEGER primary(1) secondary(2)	read-only	mandatory	Y
devCfgClkSource	{ devConfigClockSrcEntry 2 }	INTEGER internal(1) external(2) interface(3) dbm(4)	read-write	mandatory	Y
devCfgClkIfIndex	{ devConfigClockSrcEntry 3 }	INTEGER	read-write	mandatory	Y

5.26.4.3.1.1 devCfgClkWhichSrc

The clock source to which this entry is applicable. The following values are defined for this object:

- primary(1) - The primary master clock source
- secondary(2) - The secondary master clock source

5.26.4.3.1.2 devCfgClkSource

This object is used to select the source for the master clock for the device. The source selected provides synchronization for all the timing within the device, and the clocks for all of the external interfaces. The following values are defined for this object:

- internal(1) - Master clock is the internal clock
- external(2) - Master clock is the external clock source
- interface(3) - clock source is provided via an interface
- dbm(4) - clock source is provided via a backup module

5.26.4.3.1.3 devCfgClkIfIndex

This object is used to select the interface to be used as the source for the master clock for the device, if devCfgClkSource is set to interface(3). The interface selected provides synchronization for all the timing within the device, and the clocks for all of the external interfaces.

NOTE:If a synchronous data port is selected and the EDL is enabled for that synchronous data port, then the external device must provide a clock 8K bps less than the expected data port rate. For example, if the data port rate is set at 64K bps, the external clock source needs to supply a 56K bps clock signal.

If devCfgClkSource is not interface(3), the value of this object is meaningless.

Valid Object Value(s)

The NOTE above does not apply to BitStorm 2600 and GranDSLAM 4200 IP DSLAM.

5.26.4.4 devConfigTrap

The Trap Configuration group.

OIDs:

- devConfigTrap - { pdn-devConfig 4 } 1.3.6.1.4.1.1795.2.24.2.7.4

Table 245. devConfigTrap

Object	OID	Syntax	Access	Status	Supported
devConfigTrapEnable	{ devConfigTrap 1 }	INTEGER	read-write	mandatory	Y
cCNTrapEnable	{ devConfigTrap 2 }	INTEGER	read-write	mandatory	Y

5.26.4.4.1 devConfigTrapEnable

Which trap types are sent by the entity. This object is a bit map represented as a sum, therefore, multiple trap types can be enabled simultaneously. This objects sets flag (enable/disable) value on entity level. If trap enable/disable flags are set per interface level by some other object this object will read “enabled” if any one of the interfaces is enabled. If any of the traps below are set to enabled by this object it will enable that trap on all interfaces on the entity.

The various bit positions are:

Bit Position	Meaning
1	warmStart traps
2	authenticationFailure
4	enterpriseSpecific traps
8	LinkUp trap
16	LinkDown Trap

5.26.4.4.2 cCNTrapEnable

This object is used to enable or disable configuration change trap. It is a bit sequence sum with only accepted values 1 or 2. Setting the bit sequence to 2 would mean enabling the trap and 1 would mean disabling it.

5.26.4.5 devConfigAlarm

The Paradyne System Alarm group.

OIDs:

- devConfigAlarm - { pdn-devConfig 5 } 1.3.6.1.4.1.1795.2.24.2.7.5

Table 246. devConfigAlarm

Object	OID	Syntax	Access	Status	Supported
devConfigAlarmRelayCutoff	{ devConfigAlarm 1 }	INTEGER noOp(1) off(2)	read-write	mandatory	N

5.26.4.5.1 devConfigAlarmRelayCutoff

Writing off(2) to this object will turn off the System Alarm Relay. Reading this object will always return noOp(1).

This object can take on the following values:

- noOp(1)
- off(2)

5.26.4.6 devConfigCardType

The Paradyne Card Type table.

IDs:

- devConfigCardType - { pdn-devConfig 6 } 1.3.6.1.4.1.1795.2.24.2.7.6

Table 247. devConfigCardType

Object	OID	Syntax	Access	Status	Supported
devConfigCardTypeTable	{ devConfigCardType 7 }	SEQUENCE OF devConfigCardTypeEntry	not-accessible	mandatory	N

This table is used by multi-slot devices. It shows what type of card has been configured to occupy each slot in the chassis, and what card type is actually present in the chassis. If the configured and actual card types do not match, this table allows the actual card type to be accepted, which will change the configuration information for the card.

5.26.4.6.1 devConfigCardTypeTable

This table is used by multi-slot devices. It shows what type of card has been configured to occupy each slot in the chassis, and what card type is actually present in the chassis. If the configured and actual card types do not match, this table allows the actual card type to be accepted, which will change the configuration information for the card. The table is a sequence of devConfigCardTypeEntry where each entry contains configuration information regarding a card in a particulare slot.

Index:

- devCfgCardSlot

OIDs:

- devConfigCardType - { devConfigCardType 7 } 1.3.6.1.4.1.1795.2.24.2.7.6.7
- devConfigCardEntry - { devConfigCardType 1 } 1.3.6.1.4.1.1795.2.24.2.7.6.7.1

Table 248. devConfigCardTypeTable

Object	OID	Syntax	Access	Status	Supported
devCfgCardSlot	{ devConfigCardEntry 1 }	INTEGER	read-only	mandatory	N

Table 248. devConfigCardTypeTable

Object	OID	Syntax	Access	Status	Supported
devCfgCardConfig	{ devConfigCardEntry 2 }	INTEGER emptySlot(1) unsupportedAPM(2) t1NAM(3) syncDataAPM(4) voiceFxsAPM(5) voiceEmAPM(6) voiceFxoAPM(7) dsxAPM(8) t1NoDsxNAM(9) misconfiguredAPM(10) ocu2APM(11) ocu6APM(12) dce6APM(13) sruAPM(14 ocu4APM(15) pktVoiceAPM(16) acceptingAPM(17) failedAPM(18) dpNAM(19) stNAM(20) ddsNAM(21) dualDsxNniNAM(22) t3NniNAM(23) t3NAM(24) dslNAM(25)	read-only	mandatory	N
devCfgCardActual	{ devConfigCardEntry 3 }	INTEGER emptySlot(1) unsupportedAPM(2) t1NAM(3) syncDataAPM(4) voiceFxsAPM(5) voiceEmAPM(6) voiceFxoAPM(7) dsxAPM(8) t1NoDsxNAM(9) misconfiguredAPM(10) ocu2APM(11) ocu6APM(12) dce6APM(13) sruAPM(14 ocu4APM(15) pktVoiceAPM(16) acceptingAPM(17) failedAPM(18) dpNAM(19) stNAM(20) ddsNAM(21) dualDsxNniNAM(22) t3NniNAM(23) t3NAM(24) dslNAM(25)	read-only	mandatory	N
devCfgCardAction	{ devConfigCardEntry 4 }	INTEGER noOp(1) accept(2)	read-write	mandatory	N

5.26.4.6.1.1 devCfgCardSlot

The slot number which this card occupies in the chassis.

5.26.4.6.1.2 devCfgCardConfig

The type of card which has been configured for this slot. The following values are defined for this object:

- emptySlot(1) - an empty slot
- unsupportedAPM(2) - an unrecognized APM
- t1NAM(3) - a 1T1/1DSX NAM
- syncDataAPM(4) - a synchronous data port APM
- voiceFxsAPM(5) - an FXS Voice APM
- voiceEmAPM(6) - an E&M Voice APM
- voiceFxoAPM(7) - an FXO Voice APM
- dsxAPM(8) - an DSX-1 APM
- t1NoDsxNAM(9) - a 2T1 with No DSX NAM
- misconfiguredAPM(10) - a misconfigured APM
- ocu2APM(11) - a 2 port OCU APM
- ocu6APM(12) - a 6 port OCU APM
- dce6APM(13) - a 6-port DCE APM
- sruAPM(14) - a 5-port SubRate APM
- ocu4APM(15) - a 4-port Office Channel Unit APM
- pktVoiceAPM(16) - a Packet Voice APM
- acceptingAPM(17) - Accepting State
- failedAPM(18) - APM position has failed card
- dpNAM(19) - Data Port NAM
- stNAM(20) - S/T ISDN NAM
- ddsNAM(21) - DDS NAM
- dualDsxNniNAM(22) - Dual Dsx-1 NNI NAM
- t3NniNAM(23) - T3 NNI NAM
- t3NAM(24) - T3 NAM
- dslNAM(25) - DSL NAM

5.26.4.6.1.3 devCfgCardActual

The type of card which is present in this slot. The following values are defined for this object:

- emptySlot(1) - an empty slot
- unsupportedAPM(2) - an unrecognized APM
- t1NAM(3) - a 1T1/1DSX NAM
- syncDataAPM(4) - a synchronous data port APM
- voiceFxsAPM(5) - an FXS Voice APM
- voiceEmAPM(6) - an E&M Voice APM
- voiceFxoAPM(7) - an FXO Voice APM
- dsxAPM(8) - an DSX-1 APM
- t1NoDsxNAM(9) - a 2T1 with No DSX NAM
- misconfiguredAPM(10) - a misconfigured APM
- ocu2APM(11) - a 2 port OCU APM

- ocu6APM(12) - a 6 port OCU APM
- dce6APM(13) - a 6-port DCE APM
- sruAPM(14) - a 5-port SubRate APM
- ocu4APM(15) - a 4-port Office Channel Unit APM
- pktVoiceAPM(16) - a Packet Voice APM
- acceptingAPM(17) - Accepting State
- failedAPM(18) - APM position has failed card
- dpNAM(19) - Data Port NAM
- stNAM(20) - S/T ISDN NAM
- ddsNAM(21) - DDS NAM
- dualDsxNniNAM(22) - Dual Dsx-1 NNI NAM
- t3NniNAM(23) - T3 NNI NAM
- t3NAM(24) - T3 NAM
- dslNAM(25) - DSL NAM

5.26.4.6.1.4 devCfgCardAction

This object can take on the following values:

- noOp(1)
- accept(2)

Writing accept(2) to this object changes the configured card type to match the type of card currently present in the slot. Reading this object always returns noOp(1).

5.26.4.7 devConfigNetSync

The Paradyne Device Network Synchronization group.

OIDs:

- devConfigNetSync - { pdn-devConfig 7 } 1.3.6.1.4.1.1795.2.24.2.7.7

Table 249. devConfigNetSync

Object	OID	Syntax	Access	Status	Supported
devConfigNetSyncRole	{ devConfigNetSync 1 }	INTEGER none(1) tributary(2) controller(3)	read-write	mandatory	N

5.26.4.7.1 devConfigNetSyncRole

Network Time Synchronization role item for a device in the network. The following values are defined for this object:

- none(1) - Device will ignore network time synchronization messages.
- tributary(2) - Update the network reference time each time the device receives an update massage.
- controller(3) - The device will generate the synchronization message and Time of Day clock is the same as the Network Reference Time.

The default value is “tributary.”

5.26.4.8 devConfigTime

The Paradyne Device Configuration Time group.

OIDs:

- devConfigTime - { pdn-devConfig 8 } 1.3.6.1.4.1.1795.2.24.2.7.8

Table 250. devConfigTime

Object	OID	Syntax	Access	Status	Supported
devConfigTimeOfDay	{ devConfigTime 1 }	DateAndTime	read-write	mandatory	Y

This is standard definition of Device Time of Day (DateAndTime) SYNTAX.

Valid Object Value(s)

The following fields will not be supported:

- 8 - Direction form UTC
- 10 - Hours form UTC
- 11 - Minutes from UTC

The timezone information specified as an offset from UTC is set using the devNTPOffsetFromUTC MIB object, Section 5.43.3.2.5, Page 498.

The only difference is that our group is defining timezone filed (8) optional.

5.26.4.9 devConfigChangeKeys

The Paradyne Device Configuration Change Key Group

OIDs:

- devConfigChangeKeys - { pdn-devConfig 9 } 1.3.6.1.4.1.1795.2.24.2.7.9

Table 251. devConfigChangeKeys

Object	OID	Syntax	Access	Status	Supported
devConfigChangeKeysTable	{ devConfigChangeKeys 1 }	SEQUENCE OF devConfigChangeKeysEntry	not-accessible	mandatory	N

5.26.4.9.1 devConfigChangeKeysTable

The Paradyne Configuration Change Key Table contains a set of keys specific to each non-volatile database on the device. The keys are ensured to change each time there is a change to the database that they represent.

Example of Use:

1) Database Discovery

This MIB can be used to discover the various non-volatile databases on a device. Each database has a unique key.

The discovery process consists of walking devConfigChangeKeysDbKey. Each response corresponds to a different database type. The NMS only needs to be concerned with the databases that affect them. Each of the “interesting” database keys should be stored for later reference.

2) Database monitoring

After discovery, the NMS can check the database keys of the databases of interest to it either at specific intervals or based on configuration change traps. When using trap directed, polling, a default polling interval should still apply in case a trap is lost in the transmission process.

Checking the keys consists of performing a get operation on devConfigChangeKeysDbKey for the specific database(s) of interest. If the key has changed from that previously stored, the database has changed. The new key should be stored for future reference.

The Paradyne Configuration Change Key Table. is in place to allow a Network Manager to know when the configuration on the device changes. Configuration can be found in several databases that may each have a unique access method. For this reason, the keys are in the form of a table.

Index:

- devConfigChangeKeysDbType

OIDs:

- devConfigChangeKeysTable - { devConfigChangeKeys 1 } 1.3.6.1.4.1.1795.2.24.2.7.9.1
- devConfigChangeKeysEntry - { devConfigChangeKeysTable 1 } 1.3.6.1.4.1.1795.2.24.2.7.9.1.1

Table 252. devConfigChangeKeysTable

Object	OID	Syntax	Access	Status	Supported
devConfigChangeKeysDbType	{ devConfigChangeKeysEntry 1 }	INTEGER generalConfig(1) rmonAlarm(2) rmonUserHistory(3) routerConfig(4)	read-only	mandatory	N
devConfigChangeKeysDbKey	{ devConfigChangeKeysEntry 2 }	Gauge	read-only	mandatory	N

5.26.4.9.1.1 devConfigChangeKeysDbType

The type of database that is being keyed. If a specific type is not supported by the device, noSuchName should be returned. The following values are defined for this object:

- generalConfig(1)
- rmonAlarm(2)
- rmonUserHistory(3)
- routerConfig(4)

A unique value that will change each time the database is altered. Effort should be made to increase the number of times the database can change without seeing the same key.

5.26.4.9.1.2 devConfigChangeKeysDbKey

A unique value that will change each time the database is altered. Effort should be made to increase the number of times the database can change without seeing the same key.

5.26.4.10 devConfiguration

The General Configuration group.

OIDs:

- devConfiguration - { Pdn-devConfig 10 } 1.3.6.1.4.1.1795.2.24.2.7.10

Table 253. devConfiguration

Object	OID	Syntax	Access	Status	Supported
devConfigComDiscTime	{ devConfiguration 1 }	INTEGER	read-write	mandatory	Y
devConfigPortNumDisplayFormat	{ devConfiguration 2 }	INTEGER sle(1) unitport(2) name(3)	read-only[a]	mandatory	Y
devConfigDateDisplayFormat	{ devConfiguration 3 }	INTEGER ddmmyy(1) mmddyy(2)	read-write	mandatory	Y
devAcceptRemoteResetFrame	{ devConfiguration 4 }	INTEGER enable(1) disable(2)	read-write	mandatory	Y

a. Product imposed restriction

5.26.4.10.1 devConfigComDiscTime

This object is used to specify how long to wait before disconnecting. Inactivity is defined as no keyboard activity within a given period of time. The time is kept in seconds. The default is 300 seconds.

5.26.4.10.2 devConfigPortNumDisplayFormat

The following values are defined for this object:

- sle(1)
- unitport(2)
- name(3)

This object is used to set display format for the port numbers. The default is unit/port number. sle(1) - this method is to use Single Logical Entity interface numbers. For example, if sle is configured the sle numbers range from 1 to 384 for a stack of 8 units with 48 ports each. unitport(2) - this method uses unit number/port number For previous example the numbers would be from 1/1 to 8/48. name(3) - This method allows users to assign character strings (names) to each interface and to view the interfaces by names.

Valid Object Value(s)

- sle(1)

5.26.4.10.3 devConfigDateDisplayFormat

The following values are defined for this object which specify which format the date is displayed in:

- ddmmyy(1)
- mmddyy(2)

The default is mmddyy(2).

5.26.4.11 devAcceptRemoteResetFrame

This object is used to enable/disable acceptance of remote reset frame which would result in hardware reset. The following values are defined for this object:

- enable(1)
- disable(2)

The default setting is disabled.

5.26.5 Traps

OIDs:

- devConfigTrap - { pdn-devConfig 4 } 1.3.6.1.4.1.1795.2.24.2.7.4

Table 254. PDN-CONFIG-MIB Traps

Trap	ENTERPRISE	:=	Variable Bindings / Objects	Status	Supported
cCN	devConfigTrap	1	ifIndex	current	N

5.26.5.1 cCN

This trap signifies a Configuration change or software upgrade. This trap is of “warning” class 7.

5.27 PDN-CONTROL-MIB — pdn_Control.mib

5.27.1 Introduction

This MIB Module allows a user to reset a device in the DSLAM, as well as perform various operations related to the storage and retrieval of firmware and configuration files on devices.

The version of the MIB used for this document had a LAST-UPDATED of 200412140000Z which decodes to December 14, 2004.

5.27.2 Textual Conventions

No “special” textual conventions are made in this MIB.

5.27.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnControl - { pdn-common 10 } 1.3.6.1.4.1.1795.2.24.2.10
- pdnControlMIBTrapsV2 - { pdnControl 0 } 1.3.6.1.4.1.1795.2.24.2.10.0
- devHWControlReset - { pdnControl 1 } 1.3.6.1.4.1.1795.2.24.2.10.1
- devControlTestTable - { pdnControl 2 } 1.3.6.1.4.1.1795.2.24.2.10.2
- devControlDownLoadTable - { pdnControl 3 } 1.3.6.1.4.1.1795.2.24.2.10.3
- devControlRMON - { pdnControl 4 } 1.3.6.1.4.1.1795.2.24.2.10.4
- devSNSwitchFirmwareTable - { pdnControl 5 } 1.3.6.1.4.1.1795.2.24.2.10.5
- devControlFTP - { pdnControl 6 } 1.3.6.1.4.1.1795.2.24.2.10.6
- devFileXferMIBObjects - { pdnControl 7 } 1.3.6.1.4.1.1795.2.24.2.10.7
- devFileXferMIBTraps - { pdnControl 8 } 1.3.6.1.4.1.1795.2.24.2.10.8
- devFirmwareControlMIBObjects - { pdnControl 9 } 1.3.6.1.4.1.1795.2.24.2.10.9
- pdnConfigChangeMgmt - { pdnControl 10 } 1.3.6.1.4.1.1795.2.24.2.10.10
- pdnControlMIBGroups - { pdnControl 11 } 1.3.6.1.4.1.1795.2.24.2.10.11
- pdnAutoFw - { pdnControl 12 } 1.3.6.1.4.1.1795.2.24.2.10.12

5.27.4 MIB Objects

Table 255. PDN-CONTROL-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnControlMIBTrapsV2	{ pdnControl 0 }	OBJECT-IDENTITY	N/A	current	Y
devHWControlReset	{ pdnControl 1 }	INTEGER noOp(1) reset(2)	read-write	current	N
devControlTestTable	{ pdnControl 2 }	SEQUENCE OF devControlTestEntry	not-accessible	current	N
devControlDownLoadTable	{ pdnControl 3 }	SEQUENCE OF devControlDownLoad-Entry	not-accessible	current	N
devControlRMON	{ pdnControl 4 }	OBJECT IDENTIFIER	N/A	N/A	N
devSNSwitchFirmwareTable	{ pdnControl 5 }	SEQUENCE OF devSNSwitchFirmware-Entry	not-accessible	current	N
devControlFTP	{ pdnControl 6 }	OBJECT IDENTIFIER	N/A	N/A	N
devFileXferMIBObjects	{ pdnControl 7 }	OBJECT IDENTIFIER	N/A	N/A	Y
devFileXferMIBTraps	{ pdnControl 8 }	OBJECT IDENTIFIER	N/A	N/A	N
devFirmwareControlMIBObjects	{ pdnControl 9 }	OBJECT IDENTIFIER	N/A	N/A	Y
pdnConfigChangeMgmt	{ pdnControl 10 }	OBJECT IDENTIFIER	N/A	N/A	Y
pdnControlMIBGroups	{ pdnControl 11 }	OBJECT IDENTIFIER	N/A	N/A	N/A
pdnAutoFw	{ pdnControl 12 }	OBJECT IDENTIFIER	B./A	N/A	N

5.27.4.1 devHWControlReset

The following values are defined for this object:

- noOp(1)
- reset(2)

Writing the value reset(2) to this object initiates a Hardware power-on reset of the device. The value read from this object is noOp(1).

5.27.4.2 devControlTestTable

This table controls device-level tests. This table consists of a sequence of devControlTestEntry where each entry consists of objects relative to device-level tests.

Index:

- devControlTest

OIDs:

- devControlTestTable - { pdnControl 2 } 1.3.6.1.4.1.1795.2.24.2.10.2
- devControlTestEntry - { devControlTestTable 1 } 1.3.6.1.4.1.1795.2.24.2.10.2.1

Table 256. devControlTestTable

Object	OID	Syntax	Access	Status	Supported
devControlTest	{ devControlTestEntry 1 }	INTEGER lampTest(1) v35DTELpbkTest(2)	not-accessible	current	N
devControlTestStatus	{ devControlTestEntry 2 }	INTEGER active(1) inactive(2)	read-only	current	N
devControlTestCmd	{ devControlTestEntry 3 }	INTEGER start(1) stop(2)	read-create	current	N

5.27.4.2.1 devControlTest

An index into the test table that indicates which test the row involves. The following values are defined for this object:

- lampTest (1)
- v35DTELpbkTest (2)

5.27.4.2.2 devControlTestStatus

The test status on the device. This object indicates whether the indexed test is currently active(1) or inactive(2).

5.27.4.2.3 devControlTestCmd

Used to start or stop the indexed test. When read, the value returned is the next logical command. The following values are defined for this object:

- start (1)
- stop (2)

5.27.4.3 devControlDownLoadTable

This table is used to control and provide information concerning downloaded executable images. This table provides a list of the current Download Releases and their associated status. Each Download Release is indexed by a number from 1 to N. The user is able to view the Download Release String and Operational Status of the release (valid or invalid) and activate a valid Download Release by changing the Administration Status to active. This table is a sequence of devControlDownLoadEntry.

Index:

- devControlDownLoadIndex

OIDs:

- devControlDownLoadTable - { pdnControl 3 } 1.3.6.1.4.1.1795.2.24.2.10.3
- devControlDownLoadEntry - { devControlDownLoadTable 1 } 1.3.6.1.4.1.1795.2.24.2.10.3.1

Table 257. devControlDownLoadTable

Object	OID	Syntax	Access	Status	Supported
devControlDownLoadIndex	{ devControlDownLoadEntry 1 }	Integer32	not-accessible	current	N

Table 257. devControlDownLoadTable

Object	OID	Syntax	Access	Status	Supported
devControlDownLoadRelease	{ devControlDownLoadEntry 2 }	DisplayString (SIZE(1…16))	read-only	current	N
devControlDownLoad-OperStatus	{ devControlDownLoadEntry 3 }	INTEGER valid(1) invalid(2)	read-only	current	N
devControlDownLoad-AdminStatus	{ devControlDownLoadEntry 4 }	INTEGER active(1) inactive(2)	read-create	current	N

5.27.4.3.1 devControlDownLoadIndex

This object is used to index the download table (range 1 to N).

5.27.4.3.2 devControlDownLoadRelease

This object indicates the Software Release for this Download. If the Software Download is Operational Status is invalid, the Software Revision Number will be blank.

5.27.4.3.3 devControlDownLoadOperStatus

This object indicates whether or not the indexed download entry contains a valid(1) or invalid(2) download.

5.27.4.3.4 devControlDownLoadAdminStatus

The following values are defined for this object:

- active(1)
- inactive(2)

This object indicates whether or not the indexed download entry is active(1) or inactive(2). Writing active(1) will activate that software release and cause the unit to reset (response may timeout). Writing active(1) to a download entry whose devControlDownloadOperStatus is invalid will return BAD VALUE. Writing inactive(2) will always return BAD VALUE.

5.27.4.4 devControlRMON

This is the RMON Group and is a collection of items that control the RMON process.

OID:

- devControlRMON - { pdnControl 4 } 1.3.6.1.4.1.1795.2.24.2.10.4

Table 258. devControlRMON

Object	OID	Syntax	Access	Status	Supported
devControlRMONAdminStatus	{ devControlRMON 1 }	INTEGER enabled(1) disabled(2)	read-write	current	N

5.27.4.4.1 devControlRMONAdminStatus

The desired state of the RMON process. If set to enabled(1), the RMON process will perform the collection tasks it is set up to do. If it is disabled(2), RMON will not perform collection.

5.27.4.5 devSNSwitchFirmwareTable

A list of objects needed to switch bank and reset the Service Node (ASN). This table is a sequence of devSNSwitchFirmwareEntry where each entry is a list of objects needed to switch bank and reset the Service Node (ASN).

Index:

- devSNSwitchFirmwareIndex

OIDs:

- devSNSwitchFirmwareTable - { pdnControl 5 } 1.3.6.1.4.1.1795.2.24.2.10.5
- devSNSwitchFirmwareEntry - { devSNSwitchFirmwareTable 1 } 1.3.6.1.4.1.1795.2.24.2.10.5.1

Table 259. devSNSwitchFirmwareTable

Object	OID	Syntax	Access	Status	Supported
devSNSwitchFirmwareIndex	{ devSNSwitchFirmwareEntry 1 }	Integer32	not-accessible	current	N
devSNSwitchFirmwareBank	{ devSNSwitchFirmwareEntry 2 }	INTEGER noOp(1) switch(2)	read-create	current	N

5.27.4.5.1 devSNSwitchFirmwareIndex

An index into the Service Node switch firmware table that indicates which interface the row involves. This index corresponds to the entityPhysicalIndex of the entity MIB.

5.27.4.5.2 devSNSwitchFirmwareBank

The following values are defined for this object:

- noOp(1)
- switch(2)

Writing the value switch(2) to this object will switch the banks to change the firmware being used on the service node. The firmware from the stand by (not active) bank is used as the currently running firmware. The value read from this object is noOp(1).

5.27.4.6 devControlFTP

This group is a collection of items that control the FTP process.

OID:

- devControlFTP - { pdnControl 6 } 1.3.6.1.4.1.1795.2.24.2.10.6

Table 260. devControlFTP

Object	OID	Syntax	Access	Status	Supported
devControlFTPRate	{ devControlFTP 1 }	Integer32	read-write	current	N

5.27.4.6.1 devControlFTPRate

FTP max transfer rate in (kbps)

5.27.4.7 devFileXferMIBObjects

OIDs:

- devFileXferMIBObjects - { pdnControl 7 } 1.3.6.1.4.1.1795.2.24.2.10.7

Table 261. devFileXferMIBObjects

Object	OID	Syntax	Access	Status	Supported
devFileXferConfigTable	{ devFileXferMIBObjects 1 }	SEQUENCE OF devFileXferConfigEntry	not-accessible	current	N
pdnDevFileXferTable	{ devFileXferMIBObjects 2 }	SEQUENCE OF pdnDevFileXferEntry	not-accessible	current	Y
pdnDevFileXferSessionIDNext	{ devFileXferMIBObjects 3 }	Integer32	read-only	current	Y

5.27.4.7.1 devFileXferConfigTable

The Paradyne FileXfer Client Config Table. This table is a sequence of devFileXferConfigEntry where each entry is a collection of objects pertaining to a file transfer client.

Use of the File Tranfer MIB could be used with other MIBS in the following manner:

1) Use another MIB/means to verify available space /make room for a file to be transferred to this device

2) Use this MIB to download the file

3) Use another MIB/means to select the file you want to make active if your selecting firmware for example.

4) Use another MIB/means to reset the device.

A management station wishing to initiate a file transfer needs to create an entry in this table. To do so, you must first identify the ifIndex of the device you intend to do the transfer with. You should then create the associated instance of the row status It must also, either in the same or in successive PDUs, create an instance of devFileXferFileName, DevFileXferFileType, devFileXferServerIpAddress, devFileXferOperation.

It should also modify the default values for the other configuration objects if the defaults are not appropriate.

Once the appropriate instance of all the configuration objects have been created, either by an explicit SNMP set request or by default, the row status should be set to active to initiate the request. Note that this entire procedure may be initiated via a single set request which specifies a row status of createAndGo as well as specifies valid values for the non-defaulted configuration objects.

Once the DevFileXferConfigEntry request has been created (that is, the devFileXferRowStatus has been made active), the entry cannot be modified - the only operation possible after this is to delete the row.

Once the request completes, the management station should retrieve the values of the status objects of interest, and should then delete the entry. In order to prevent old entries from clogging the table, entries could be aged out, but an entry will never be deleted within 5 minutes of completing.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- devFileXferConfigTable - { devFileXferMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.10.7.1
- devFileXferConfigEntry - { devFileXferConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.10.7.1.1

Table 262. devFileXferConfigTable

Object	OID	Syntax	Access	Status	Supported
devFileXferFileName	{ devFileXferConfigEntry 1 }	DisplayString	read-create	current	N
devFileXferCopyProtocol	{ devFileXferConfigEntry 2 }	INTEGER tftp(1) ftp(2)	read-create	current	N
devFileXferFileType	{ devFileXferConfigEntry 3 }	INTEGER firmware(1) config(2)	read-create	current	N
devFileXferServerIpAddress	{ devFileXferConfigEntry 4 }	IpAddress	read-create	current	N
devFileXferUserName	{ devFileXferConfigEntry 5 }	DisplayString	read-create	current	N
devFileXferUserPassword	{ devFileXferConfigEntry 6 }	DisplayString	read-create	current	N
devFileXferOperation	{ devFileXferConfigEntry 7 }	INTEGER get(1) put(2)	read-create	current	N
devFileXferPktsSent	{ devFileXferConfigEntry 8 }	Counter32	read-only	current	N
devFileXferPktsRecv	{ devFileXferConfigEntry 9 }	Counter32	read-only	current	N
devFileXferOctetsSent	{ devFileXferConfigEntry 9 }	Counter32	read-only	current	N
devFileXferOctetsRecv	{ devFileXferConfigEntry 11 }	Counter32	read-only	current	N
devFileXferOwnerString	{ devFileXferConfigEntry 12 }	OCTET STRING (SIZE(0…255))	read-create	current	N
devFileXferStatus	{ devFileXferConfigEntry 13 }	INTEGER none(1) success(2) failure(3) inprogress(4)	read-only	current	N
devFileXferErrorStatus	{ devFileXferConfigEntry 14 }	Integer32	read-only	current	N
devFileXferSendEvent	{ devFileXferConfigEntry 15 }	INTEGER active(1) inactive(2)	read-create	current	N
devFileXferRowStatus	{ devFileXferConfigEntry 16 }	RowStatus	read-create	current	N
devFileXferXferTime	{ devFileXferConfigEntry 17 }	TimeTicks	read-only	current	N

5.27.4.7.1.1 devFileXferFileName

This object contains the name of the file transfer file.

5.27.4.7.1.2 devFileXferCopyProtocol

The following values are defined for this object:

- tftp(1)
- ftp(2)

The transfer protocol that should be used to copy the file across the network. If the file transfer is to occur locally on the SNMP agent, the method of transfer is left up to the implementation, and is not restricted to the protocols below.

The default value is ftp.

5.27.4.7.1.3 devFileXferFileType

Specifies the type of file your want to transfer. The following values are defined for this object:

- firmware(1)
- config(2)

5.27.4.7.1.4 devFileXferServerIpAddress

This object contains the file transfer server's IP address.

5.27.4.7.1.5 devFileXferUserName

This object contains the username if a username is needed to login to the server.

5.27.4.7.1.6 devFileXferUserPassword

This object contains the password if a password is needed to login to the server. This object will return null on a read operation.

5.27.4.7.1.7 devFileXferOperation

This object contains the operation the file transfer wants to perform. The following values are defined for this object:

- get(1)
- put(2)

5.27.4.7.1.8 devFileXferPktsSent

This object contains the number of packets sent to the server at the time of interrogation.

5.27.4.7.1.9 devFileXferPktsRecv

This object contains the number of packets received from the server at the time of interrogation.

5.27.4.7.1.10 devFileXferOctetsSent

This object contains the number of octets sent to the server at the time of interrogation.

5.27.4.7.1.11 devFileXferOctetsRecv

This object contains the number of octets received from the server at the time of interrogation.

5.27.4.7.1.12 devFileXferOwnerString

The entity which currently has the "ownership" required to invoke the operation on this index.

5.27.4.7.1.13 devFileXferStatus

This object contains the status of the file transfer. The following values are defined for this object:

- none(1)
- success(2)
- failure(3)
- inprogress(4)

5.27.4.7.1.14 devFileXferErrorStatus

This object contains the reason code of the failure determined in devFileXferStatus. The reason code is specific to the file transfer protocol. Please refer to the file transfer protocols respective RFC for clarification of the error code value meanings:

- TFTP Error Codes from RFC 1350
- FTP Error Codes from RFC 959

5.27.4.7.1.15 devFileXferSendEvent

This object indicates whether or not to send the devFileXferEvent event trap. The following values are defined for this object:

- active(1)
- inactive(2)

5.27.4.7.1.16 devFileXferRowStatus

This object is used to create a new row or delete an existing row in this table.

5.27.4.7.1.17 devFileXferXferTime

This object indicates the elapsed time (in hundredths of a second) of the file transfer.

5.27.4.7.2 pdnDevFileXferTable

The Paradyne FileXfer Image Client Config Table accessed by session id. This table is a sequence of pdnDevFileXferEntry where each entry is a collection of objects pertaining to a file transfer client.

Use of the File Tranfer MIB could be used with other MIBS in the following manner:

1) Use another MIB/means to verify available space /make room for a file to be transferred to this device

2) Use this MIB to download the file

3) Use another MIB object/means to select the file you want to make active if your selecting firmware for example.

4) Use another MIB/means to reset the device.

A management station wishing to initiate a file transfer needs to create an entry in this table. To do so, you must first identify the session id of the session that the transfer is done with. This is done by setting SessionID to 0 on SET and when session is establish entering that session id into this object. You should then create the associated instance of the row status It must also, either in the same or in successive PDUs, create an instance of pdnDevFileXferFileName, pdnDevFileXferFileType, pdnDevFileXferServerIpAddress, pdnDevFileXferOperation.

It should also modify the default values for the other configuration objects if the defaults are not appropriate.

Once the appropriate instance of all the configuration objects have been created, either by an explicit SNMP set request or by default, the row status should be set to active to initiate the request. Note that this entire procedure may be initiated via a single set request which specifies a row status of createAndGo as well as specifies valid values for the non-defaulted configuration objects.

Once the request completes, the management station should retrieve the values of the status objects of interest, and should then delete the entry. In order to prevent old entries from clogging the table, entries could be aged out, but an entry will never be deleted within 5 minutes of completing.

Index:

- pdnDevFileXferSessionID

OIDs:

- pdnDevFileXferTable - { devFileXferMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.10.7.2
- pdnDevFileXferEntry - { pdnDevFileXferTable 1 } 1.3.6.1.4.1.1795.2.24.2.10.7.2.1

Table 263. pdnDevFileXferTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
pdnDevFileXferSessionID	{ pdnDevFileXferEntry 1 }	Integer32	not-accessible	current	Y
pdnDevFileXferifIndex	{ pdnDevFileXferEntry 2 }	Integer32	read-create	current	Y
pdnDevFileXferFileName	{ pdnDevFileXferEntry 3 }	DisplayString	read-create	current	Y

Table 263. pdnDevFileXferTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
pdnDevFileXfer-CopyProtocol	{ pdnDevFileXferEntry 4 }	INTEGER tftp(1) ftp(2)	read-create	current	Y
pdnDevFileXferFileType	{ pdnDevFileXferEntry 5 }	INTEGER firmware(1) config(2)	read-create	current	Y
pdnDevFileXfer-ServerIpAddress	{ pdnDevFileXferEntry 6 }	IpAddress	read-create	current	Y
pdnDevFileXferUserName	{ pdnDevFileXferEntry 7 }	DisplayString	read-create	current	Y
pdnDevFileXfer-UserPassword	{ pdnDevFileXferEntry 8 }	DisplayString	read-create	current	Y
pdnDevFileXferUserAccount	{ pdnDevFileXferEntry 9 }	DisplayString	read-create	current	Y
pdnDevFileXferOperation	{ pdnDevFileXferEntry 10 }	INTEGER get(1) put(2)	read-create	current	Y
pdnDevFileXferPktsSent	{ pdnDevFileXferEntry 11 }	Counter32	read-only	current	Y
pdnDevFileXferPktsRecv	{ pdnDevFileXferEntry 12 }	Counter32	read-only	current	Y
pdnDevFileXferOctetSent	{ pdnDevFileXferEntry 13 }	Counter32	read-only	current	Y
pdnDevFileXferOctesRecv	{ pdnDevFileXferEntry 14 }	Counter32	read-only	current	Y
pdnDevFileXferOwnerString	{ pdnDevFileXferEntry 15 }	OCTET STRING (SIZE(1…255)	read-create	current	Y
pdnDevFileXferStatus	{ pdnDevFileXferEntry 16 }	INTEGER none(1) success(2) failure(3) inprogress(4)	read-only	current	Y
pdnDevFileXferApply	{ pdnDevFileXferEntry 17 }	INTEGER yes(1) no(2)	read-create	current	Y
pdnDefFileXferErrorStatus	{ pdnDevFileXferEntry 18 }	Integer32	read-only	current	Y
pdnDevFileXferSendEvent	{ pdnDevFileXferEntry 19 }	INTEGER active(1) inactive(2)	read-create	current	Y
pdnDevFileXferXferTime	{ pdnDevFileXferEntry 20 }	TimeTicks	read-only	current	Y
pdnDevFileXferRowStatus	{ pdnDevFileXferEntry 21 }	RowStatus	read-create	current	Y
pdnDevFileXfer-ErrorStatusText	{ pdnDevFileXferEntry 22 }	DisplayString	read-only	current	Y

5.27.4.7.2.1 pdnDevFileXferSessionID

This object is the session id. This object is used by FileXfer Table to identify the row of this table. When creating the new row in the table the value of this object may be obtained by retrieving the value of pdnDevFileXferSessionIDNext.

5.27.4.7.2.2 pdnDevFileXferifIndex

This object specifies the interface number that corresponds to RFC 1213. It this object is not used the value of this object should be zero.

5.27.4.7.2.3 pdnDevFileXferFileName

This object contains the name of the file transfer file.

5.27.4.7.2.4 pdnDevFileXferCopyProtocol

The transfer protocol that should be used to copy the file across the network. If the file transfer is to occur locally on the SNMP agent, the method of transfer is left up to the implementation, and is not restricted to the protocols below.

The following values are defined for this object:

- tftp(1)
- ftp(2)

The default value is tftp.

5.27.4.7.2.5 pdnDevFileXferFileType

This object specifies the type of file you want to transfer. The following values are defined for this object:

- firmware(1)
- config(2)

5.27.4.7.2.6 pdnDevFileXferServerIpAddress

This object contains the file transfer server's IP address.

5.27.4.7.2.7 pdnDevFileXferUserName

This object contains the username if a username is needed to login to the server.

5.27.4.7.2.8 pdnDevFileXferUserPassword

This object contains the password if a password is needed to login to the server. This object will return null on a read operation.

5.27.4.7.2.9 pdnDevFileXferUserAccount

This object contains the name of the user account information that is stored on ftp server. Server would contain user name and password under this account name. This object will return null on a read operation. The default value of this is a null string.

5.27.4.7.2.10 pdnDevFileXferOperation

This object contains the operation the file transfer wants to perform. The following values are defined for this object:

- get(1)
- put(2)

5.27.4.7.2.11 pdnDevFileXferPktsSent

This object contains the number of packets sent to the server at the time of interrogation.

5.27.4.7.2.12 pdnDevFileXferPktsRecv

This object contains the number of packets received from the server at the time of interrogation.

5.27.4.7.2.13 pdnDevFileXferOctetsSent

This object contains the number of octets sent to the server at the time of interrogation.

5.27.4.7.2.14 pdnDevFileXferOctetsRecv

This object contains the number of octets received from the server at the time of interrogation.

5.27.4.7.2.15 pdnDevFileXferOwnerString

The entity which currently has the “ownership” required to invoke the operation on this index.

5.27.4.7.2.16 pdnDevFileXferStatus

This object contains the status of the file transfer. The following values are defined for this object:

- none(1)
- success(2)
- failure(3)
- inprogress(4)

5.27.4.7.2.17 pdnDevFileXferApply

This object is configured to specify if the firmware download is applied immediately after file transfer or not. The following values are defined for this object:

- yes(1)
- no(2)

5.27.4.7.2.18 pdnDevFileXferErrorStatus

This object contains the reason code of the failure determined in devFileXferStatus. The reason code are specific to the file transfer protocol. Please refer to the file transfer protocols respective RFC for clarification of the error code value meanings. TFTP Error Codes from rfc 1350 FTP Error Codes from rfc 959

5.27.4.7.2.19 pdnDevFileXferSendEvent

This object indicates whether or not to send the pdnDevFileXferEvent event trap. The following values are defined for this object:

- active(1)
- inactive(2)

5.27.4.7.2.20 pdnDevFileXferXferTime

This object indicates the elapsed time (in hundredths of a second) of the file transfer.

5.27.4.7.2.21 pdnDevFileXferRowStatus

This object is used to create a new row or delete an existing row in this table.

5.27.4.7.2.22 pdnDevFileXferErrorStatusText

This object provides a textual interpretation of the pdnDevFileXferErrorStatus code. This is important for those situations where FTP may fail and a standard error code doesn’t apply. In the situations where standard error codes apply, this object will make it easier for an operator to react to an FTP failure.

5.27.4.7.3 pdnDevFileXferSessionIDNext

This object contains an appropriate value to be used for pdnDevFileXferTable. The value of 0 indicates that no unassigned entries are available. To obtain pdnDevFileXferSessionID value for the new entry the manager issues a management protocol retrieval operation to obtain a current value of this object. After each retrieval, the agent should modify the value to the next unassigned index.

5.27.4.8 devFirmwareControlMIBObjects

OIDs:

- devFirmwareControlMIBObjects - { pdnControl 9 } 1.3.6.1.4.1.1795.2.24.2.10.9

Table 264. devFirmwareControlMIBObjects

Object	OID	Syntax	Access	Status	Supported
devFirmware-ControlTable	{ devFirmwareControlMIBObjects 1 }	SEQUENCE OF devFirmwareControlEntry	not-accessible	current	Y

5.27.4.8.1 devFirmwareControlTable

The Firmware Control Table is used to control and provide information concerning firmware/executable images. This table is a sequence of devFirmwareControlEntry where each entry is a list of the current Firmware Releases and their associated status. Each Firmware Release is indexed by a number from 1 to N. The user is able to view the Firmware Release String and Operational Status of the release (valid or invalid) and activate a valid Firmware Release by changing the Administration Status to active.

Index:

- devFirmwareControlIndex

OIDs:

- devFirmwareControlTable - { devFirmwareControlMIBObjects 1 }
- devFirmwareControlEntry - { devFirmwareControlTable 1 }

Table 265. devFirmwareControlTable

Object	OID	Syntax	Access	Status	Supported
devFirmwareControlIndex	{ devFirmwareControlEntry 1 }	Integer32	not-accessible	current	Y
devFirmwareControlRelease	{ devFirmwareControlEntry 2 }	DisplayString (SIZE(1…16))	read-only	current	Y
devFirmwareControl-OperStatus	{ devFirmwareControlEntry 3 }	Integer valid(1) invalid(2) unknown(3)	read-only	current	Y
devFirmwareControl-AdminStatus	{ devFirmwareControlEntry 4 }	INTEGER active(1) inactive(2)	read-create	current	Y

5.27.4.8.1.1 devFirmwareControlIndex

This object is used to index the Firmware table (range 1 to N).

5.27.4.8.1.2 devFirmwareControlRelease

This object indicates the Software Release for this Firmware. If the Software Firmware is Operational Status is invalid, the Software Revision Number will be blank.

5.27.4.8.1.3 devFirmwareControlOperStatus

This object indicates whether or not the indexed Firmware entry contains a valid(1) or invalid(2) Firmware. The following values are defined for this object:

- valid(1)
- invalid(2)

- unknown(3)

5.27.4.8.1.4 devFirmwareControlAdminStatus

This object indicates whether or not the indexed Firmware entry is active(1) or inactive(2). Writing active(1) will activate that software release and cause the unit to reset (response may timeout). Writing active(1) to a Firmware entry whose devFirmwareControlOperStatus is invalid will return BAD VALUE. Writing inactive(2) will always return BAD VALUE.

5.27.4.9 pdnConfigChangeMgmt

The Configuration Change Management (CCM) group contains objects for controlling automated backup and restore of the configuration files of devices (cards) in the slots of the Paradyne DSLAM chassis.

OID:

- pdnConfigChangeMgmt - { pdnControl 10 } 1.3.6.1.4.1.1795.2.24.2.10.10

Table 266. pdnConfigChangeMgmt

Object	OID	Syntax	Access	Status	Supported
pdnCCMAutoBackup	{ pdnConfigChangeMgmt 1 }	OID	N/A	N/A	Y
pdnCCMAutoRestore	{ pdnConfigChangeMgmt 2 }	SwitchState	read-write	current	Y
pdnCCMResyncOperation	{ pdnConfigChangeMgmt 3 }	INTEGER noOp(1) get(2) put(3)	read-write	current	Y
pdnCCMOperation	{ pdnConfigChangeMgmt 4 }	INTEGER noOp(1) apply(2) save(3) reset(4) revert(5) default(6)	read-write	current	Y

5.27.4.9.1 pdnCCMAutoBackup

The CCM Auto Backup Group.

OID

- pdnCCMAutoBackup - { pdnConfigChangeMgmt 1 } 1.3.6.1.4.1.1795.2.24.2.10.10.1

Table 267. pdnCCMAutoBackup

Object	OID	Syntax	Access	Status	Supported
pdnCCMAutoBackupType	{ pdnCCMAutoBackup 1 }	INTEGER disabled(0) fixed(1) dynamic(2)	read-write	current	Y

Table 267. pdnCCMAutoBackup

Object	OID	Syntax	Access	Status	Supported
pdnCCMAutoBackupFixedDay	{ pdnCCMAutoBackup 2 }	BITS sonday(0) monday(1) tuesday(2) wednesday(3) thursday(4) firday(5) saturday(6)	read-write	current	Y
pdnCCMAutoBackupFixedTime	{ pdnCCMAutoBackup 3 }	Integer32(0…1440)	read-write	current	Y
pdnCCMAutoBackupDynamicTime	{ pdnCCMAutoBackup 4 }	Integer32(30…1440)	read-write	current	Y
pdnCCMAutoBackup-AppendTimeStampToFilename	{ pdnCCMAutoBackup 5 }	INTEGER enable(1) disable(2)	read-write	current	Y
pdnCCMAutoBackupFilename	{ pdnCCMAutoBackup 6 }	DisplayString	read-write	current	Y
pdnCCMAutoBackupServerIpAddress	{ pdnCCMAutoBackup 7 }	IpAddress	read-write	current	Y
pdnCCMAutoBackupUserName	{ pdnCCMAutoBackup 8 }	DisplayString	read-write	current	Y
pdnCCMAutoBackupUserPassword	{ pdnCCMAutoBackup 9 }	DisplayString	read-write	current	Y
pdnCCMAutoBackupUserAccount	{ pdnCCMAutoBackup 10 }	DisplayString	read-write	current	Y
pdnCCMAutoBackupCopyProtocol	{ pdnCCMAutoBackup 11 }	INTEGER tftp(1) ftp(2)	read-write	current	Y

5.27.4.9.1.1 pdnCCMAutoBackupType

This object indicates whether automatic backup of device configuration files to the MCC Flash File System is disabled, performed according to a fixed schedule, or whether it occurs dynamically at a specified interval after the latest configuration change to a device. The following values are defined for this object:

- isabled(0)
- fixed(1)
- dynamic(2)

5.27.4.9.1.2 pdnCCMAutoBackupFixedDay

This object indicates the days of the week on which fixed schedule automatic backups occur. If all bits are set, backups occur daily. The following values are defined for this object:

- sunday(0)
- monday(1)
- tuesday(2)
- wednesday(3)
- thursday(4)
- friday(5)
- saturday(6)

5.27.4.9.1.3 pdnCCMAutoBackupFixedTime

This object indicates the time of day (in minutes) past midnight, at which automatic backups on a fixed schedule occur.

5.27.4.9.1.4 pdnCCMAutoBackupDynamicTime

This object indicates the interval (in minutes) after the most recent configuration change, when a card's configuration will be backed up to the MCC Flash File System, when automatic backups are occurring on a dynamic schedule. The minimum interval is 30 minutes. The maximum interval is 24 hours (1440 minutes).

5.27.4.9.1.5 pdnCCMAutoBackupAppendTimeStampToFilename

This object indicates that the time is to be appended to filename of the file stored to upon automated backup. The following values are defined for this object:

- enable(1),
- disable(2)

The default value for this object is disabled(2).

5.27.4.9.1.6 pdnCCMAutoBackupFilename

This is file name that the configuration is stored to upon automated backup.

5.27.4.9.1.7 pdnCCMAutoBackupServerIpAddress

This object is used to configure Ip address of the server used to perform autobackup.

5.27.4.9.1.8 pdnCCMAutoBackupUserName

This object is the user name used to start ftp session for automatic backup.

5.27.4.9.1.9 pdnCCMAutoBackupUserPassword

This object is the user password that is paired with user name for ftp server login.

5.27.4.9.1.10 pdnCCMAutoBackupUserAccount

This object is the user account name that ftp server contains. Under this name ftp server stores UserName and UserPassword info. If this object is null(not configured) than the pdnCCMAutoBackupUserName and pdnCCMAutoBackupUserPassword are used.

The default is the null string.

5.27.4.9.1.11 pdnCCMAutoBackupCopyProtocol

The transfer protocol that should be used to copy the file across the network. If the file transfer is to occur locally on the SNMP agent, the method of transfer is left up to the implementation, and is not restricted to the protocols below.

The following values are defined for this object:

- tftp(1)
- ftp(2)

The default value is tftp(1).

5.27.4.9.2 pdnCCMAutoRestore

This object specifies whether the MCC Flash File System will restore stored configuration files to newly inserted cards in the DSLAM.

5.27.4.9.3 pdnCCMResyncOperation

This object contains the operation performed when ResyncState is enabled. The following values are defined for this object:

- noOp(1) - queries to this object always return noOp(1)

- get(2) - retrieves configuration files from the non-volatile to all the devices
- put(3) - stores all the configuration files on the non-volatile

5.27.4.9.4 pdnCCMOperation

This object contains configuration options for operations performed to manage object. The following values are defined for this object:

- noOp(1) - Querying this object always retrieves the value noOp(1).
- apply(2) - Performed to make configuration parameters take in effect.
- save(3) - Interpreted as copying running configuration to database (nvram).
- reset (4) - Designed to upload database (nvram), saved, configuration, and make it running configuration.
- revert (5) - Configuration option used in a case of error upon set operation (before DD applies configuration parameter(s)). It will effectively make previous applied configuration parameter set valid, running configuration. In some systems this is an internal (user transparent) operation in case of error deduced by DD.
- default(6) - Will upload default parameters to running configuration without changing saved configuration in a database (nvram).

5.27.4.10 pdnAutoFw

This is the Auto Firmware Download Group. This group contains objects for controlling automated Firmware download by MCC to the other cards in the slots of the Paradyne DSLAM chassis.

OID

- pdnAutoFw - { pdnControl 12 } 1.3.6.1.4.1.1795.2.24.2.10.12

Table 268. pdnAutoFw

Object	OID	Syntax	Access	Status	Supported
devIsAutoFwEnabled	{ pdnAutoFw 1 }	SwitchState	read-write	current	N
devAutoFwStatus	{ pdnAutoFw 2 }	DisplayString(SIZE(1…255))	read-only	current	N

5.27.4.10.1 devIsAutoFwEnabled

Writing the value enabled(1) or disabled(2) to this object enables or disables the Automatic Firmware download feature.

5.27.4.10.2 devAutoFwStatus

This indicates the Status of Auto Firmware Download where each field is separated by a semi-colon. The get function will return the status of the latest card downloaded.

5.27.5 Traps

The traps for the device control MIB.

OID

- pdnControlMIBTrapsV2 - { pdnControl 0 } 1.3.6.1.4.1.1795.2.24.2.10.0

Table 269. PDN-CONTROL-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
devFileXferEvent	{ pdnControlMIBTrapsV2 1 }	devFileXferStatus devFileXferErrorStatus devFileXferOperation devFileXferFileType devFileXferFileName	current	N
devAutoBackupFailEvent	{ pdnControlMIBTrapsV2 2 }	entPhysicalIndex	current	N
devConfigRestoreFailEvent	{ pdnControlMIBTrapsV2 3 }	entPhysicalIndex	current	N
devAutoFwEvent	{ pdnControlMIBTrapsV2 4 }	devAutoFwStatus	current	N
pdnDevFileXferEvent	{ pdnControlMIBTrapsV2 5 }	pdnDevFileXferStatus pdnDevFileXferErrorStatus pdnDevFileXferOperation pdnDevFileXferFileType pdnDevFileXferFileName pdnDevFileXferErrorStatusText	current	Y

5.27.5.1 devFileXferEvent

This trap is to communicate a couple of things about the completion of a file transfer.

- devFileXferStatus - Did it complete successfully or not
- devFileXferErrorStatus - If not, what was the error code
- devFileXferOperation - What operation was performed
- devFileXferFileType - Was it a firmware transfer or config
- devFileXferFileName - The name of the file transferred

5.27.5.2 devAutoBackupFailEvent

This trap indicates a condition exists, which prevents automated backup of a device's configuration to the Management Controller Card (MCC) Flash File System at a scheduled time.

5.27.5.3 devConfigRestoreFailEvent

This trap is sent by a device to indicate that an incorrect or incompatible configuration file is being downloaded to the device from the MCC Flash File System.

5.27.5.4 devAutoFwEvent

This trap is to communicate the result of the Auto Firmware Download. The variable binding for this trap would be the devAutoFwStatus of pdnAutoFwGroup.

5.27.5.5 pdnDevFileXferEvent

This trap is to communicate a couple of things about the completion of a file transfer.

- pdnDevFileXferStatus - Did it complete successfully or not
- pdnDevFileXferErrorStatus - If not, what was the error code
- pdnDevFileXferOperation - What operation was performed
- pdnDevFileXferFileType - Was it a firmware xfer or config

- pdnDevFileXferFileName - The name of the file transferred
- pdnDevFileXferErrorStatusText - Textual interpretation of the error code

5.28 PDN-DOT1QEXT-MIB — PDN-DOT1QEXT-MIB.mib

5.28.1 Introduction

This MIB provides additional DOT1q configuration objects not provided by RFC 2674. The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of "200211130000Z," or November 13, 2002.

5.28.2 Textual Conventions

No "special" textual conventions are made or referenced in this MIB.

5.28.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-dot1q - { pdn-common 39 } 1.3.6.1.4.1.1795.2.24.2.39
- pdnDot1qExt - { pdn-dot1q 1 } 1.3.6.1.4.1.1795.2.24.2.39.1
- pdnDot1qExtObjects - { pdnDot1qExt 1 } 1.3.6.1.4.1.1795.2.24.2.39.1.1

5.28.4 MIB Objects

Table 270. PDN-DOT1QEXT-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnDot1qVlanStaticTable	{ pdnDot1qExtObjects 1 }	SEQUENCE of pdnDot1qVlanStaticEntry	not-accessi-ble	current	Y

5.28.4.1 pdnDot1qVlanStaticExtTable

The Paradyne dot1qVlanStatic extension table. This table is used for configuring extensions to static vlans.

Augments:

- dot1qVlanStaticTable (Section 5.17.4.4.3, Page 292)

Index:

- dot1qVlanIndex (Section 5.17.4.4.2.2, Page 291)

OIDs:

- pdnDot1qVlanStaticExtTable - { pdnDot1qExtObjects 1 } 1.3.6.1.4.1.1795.2.24.2.39.1.1
- pdnDot1qVlanStaticExtEntry - { pdnDot1qVlanStaticExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.39.1.1.1

Table 271. pdnDot1qVlanStaticExtTable

Object	OID	Syntax	Access	Status	Supported
pdnDot1qVlanStaticSecureModeStatus	{ pdnDot1qVlanStaticExtEntry 1 }	INTEGER enable(1) disable(2)	read-write	current	Y
pdnDot1qVlanStaticProxyArpStatus	{ pdnDot1qVlanStaticExtEntry 2 }	INTEGER enable(1) disable(2)	read-write	current	Y
pdnDot1qVlanStaticUplink	{ pdnDot1qVlanStaticExtEntry 3 }	Integer32	read-write	current	N
pdnDot1qVlanStaticDefaultNHR	{ pdnDot1qVlanStaticExtEntry 4 }	IpAddress	read-write	current	Y

5.28.4.1.1 pdnDot1qVlanStaticSecureModeStatus

The following values are defined for this object:

- enable(1)
- disable(2)

This is objects configure secure mode status for the selected VLAN. The secure VLAN mode implies that the users on this VLAN cannot talk to users on a different VLAN. This setting also validates the configuration of the next three objects in this table. If the pdnDot1qVlanStaticSecureModeStatus is disabled non of the other three objects configurations apply.

5.28.4.1.2 pdnDot1qVlanStaticProxyArpStatus

The following values are defined for this object:

- enable(1)
- disable(2)

This objects enables/disabled proxy arp on secure VLAN. If the VLAN is secure, it can optionally have Proxy ARP enabled. This object is not valid unless pdnDot1qVlanStaticSecureModeStatus is set to ENABLED.This object is disabled(1) by default.”

5.28.4.1.3 pdnDot1qVlanStaticUplink

This is used to configure the port number of the port on which all frames belonging to secure VLAN are to be forwarded. This object is not valid unless pdnDot1qVlanStaticSecureModeStatus is set to ENABLED. If pdnDot1qVlanStaticSecureModeStatus is disabled this object will read 0 port.

5.28.4.1.4 pdnDot1qVlanStaticDefaultNHR

This object is used to configure Default Next Hop Router Ip Address that would be used only if the Next Hop Router cannot be learned (i.e from snooping messages). This object is not valid unless pdnDot1qVlanStaticSecureModeStatus is set to ENABLED. If pdnDot1qVlanStaticSecureModeStatus is disabled this object will read 0.0.0.0 address.

5.28.5 Conformance Information

5.28.6 Compliance

5.28.6.1 pdnDot1qExtCompliance

Compliance statement for using this MIB for configuring Dot1q Static Vlan table.

Optional Groups:

- pdnDot1qVlanExtGroup - Implementation of this group is required for systems wishing to conform to this MIB for the purposes of configuring Static Vlan table.

Objects with a MIN-ACCESS defined:

- pdnDot1qVlanStaticSecureModeStatus - MIN-ACCESS read-only
  The ability to set pdnDot1qVlanStaticSecureModeStatus is not required.
- pdnDot1qVlanStaticProxyArpStatus - MIN-ACCESS read-only
  The ability to set pdnDot1qVlanStaticProxyArpStatus is not required.
- pdnDot1qVlanStaticUplink - MIN-ACCESS read-only
  The ability to set pdnDot1qVlanStaticUplink is not required.
- pdnDot1qVlanStaticDefaultNHR - MIN-ACCESS read-only
  The ability to set pdnDot1qVlanStaticDefaultNHR is not required.

5.28.7 Conformance

Table 272. PDN-DOT1QEXT-MIB Conformance

Group	Description	Objects	Supported
pdnDot1qVlanExtGroup	A collection of configuration objects required for configuring Static Vlan Table.	pdnDot1qVlanStaticSecureModeStatus	Y
		pdnDot1qVlanStaticProxyArpStatus	Y
		pdnDot1qVlanStaticUplink	Y
		pdnDot1qVlanStaticDefaultNHR	Y

5.29 PDN-DSLAM-SYSTEM-MIB — pdn_dslam.mib

5.29.1 Introduction

The DSLAM MIB. This MIB is specific to port cards and DSLAM configuration/stats for MCC and the port cards. The objects defined in this mib are specific to some products and do not apply to all paradyne devices.

The version of the MIB used for this document had a LAST-UPDATED of 0206050000Z which decodes to June 5, 2002.

5.29.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- IdslClockMode - This object indicates the network clock mode set for the secondary network clock. portCardDriveClockOnboard(4), both driving the backplane and using the clock for the other Local Timing transceivers on the card. Port cards with only LT ports configured will receive a clock from the backplane using portCardSinkClock(2). portCardDriveClock(3) will drive the backplane alone.
  - — SYNTAX INTEGER
    - triState(1)
    - portCardSinkClock(2)
    - portCardDriveClock(3)
    - portCardDriveClockOnboard(4)

5.29.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-dslam - { pdn-common 24 } 1.3.6.1.4.1.1795.2.24.2.24
- sysDevDslamMIBObjects - { pdn-dslam 1 } 1.3.6.1.4.1.1795.2.24.2.24.1
- sysDevStats - { sysDevDslamMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.1
- sysDevConfig - { sysDevDslamMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.24.1.2
- sysDevDslamMIBTraps - { pdn-dslam 2 } 1.3.6.1.4.1.1795.2.24.2.24.2

5.29.4 MIB Objects

Table 273. PDN-DSLAM-SYSTEM-MIB Objects

Object	OID	Description	Supported
sysDevStats	{ sysDevDslamMIBObjects 1 }	The Statistics Group	N
sysDevConfig	{ sysDevDslamMIBObjects 2 }	The Configuration Group	Y

5.29.4.1 sysDevStats

The statistics group.

OIDs:

- sysDevStats - { sysDevDslamMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.1

Table 274. sysDevStats

Object	OID	Syntax	Access	Status	Supported
loginHistTable	{ sysDevStats 1 }	SEQUENCE OF loginHistTableEntry	not-accessible	current	Y
loginFailureCountTable	{ sysDevStats 2 }	SEQUENCE OF loginFailureCuntTableEntry	not-accessible	current	N

5.29.4.1.1 loginHistTable

This table shows the most recent 10 logins and all active users currently accessing the device. It is indexed by userId and loginTime. This table is a sequence of loginHistEntry where each entry contains information pertaining to a particular user at a particular login time.

Indices:

- loginUserId
- loginTime

OIDs:

- loginHistTable - { sysDevStats 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.1.1
- loginHistTableEntry - { loginHistTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.1.1.1

Table 275. loginHistTable

Object	OID	Syntax	Access	Status	Supported
loginUserId	{ loginHistTableEntry 1 }	DisplayString	read-only	current	Y
loginTime	{ loginHistTableEntry 2 }	TimeTicks	read-only	current	Y
loginAccessApp	{ loginHistTableEntry 3 }	INTEGER console (1) telnet (2) ftp (3) web (4) modem (5)	read-only	current	Y
loginAccessHost	{ loginHistTableEntry 4 }	IpAddress	read-only	current	Y
loginUserPriv	{ loginHistTableEntry 5 }	INTEGER administrator (1) operator (2)	read-only	current	Y
loginStatus	{ loginHistTableEntry 6 }	INTEGER active (1) inactive (2)	read-only	current	Y

5.29.4.1.1.1 loginUserId

This object contains the user login id.

5.29.4.1.1.2 loginTime

This object contains the login time in seconds when the login session is established.

5.29.4.1.1.3 loginAccessApp

This object describes the access application used by the end user to access the device. This can be done through console, using telnet, modem, web, or using ftp. The following values are defined for this object:

- console (1)
- telnet (2)
- ftp (3)
- web (4)
- modem (5)

5.29.4.1.1.4 loginAccessHost

This object contains the ip address of the network management station when the access application is telnet or ftp. In case of console, this object contains 0.0.0.0.

5.29.4.1.1.5 loginUserPriv

This object contains the access privileges of the user. The following values are defined for this object:

- administrator (1)
- operator (2)

5.29.4.1.1.6 loginStatus

This object indicates whether the user is still accessing the device. The following values are defined for this object:

- active (1)
- inactive (2)

5.29.4.1.2 loginFailureCountTable

This table contains the statistics for login failures. It is indexed by access type i.e console, telnet or ftp. This table is a sequence of loginFailureCountTableEntry where an entry corresponds to a login failure.

Index:

- loginFailureAccessApp

OIDs:

- loginFailureCountTable - { sysDevStats 2 } 1.3.6.1.4.1.1795.2.24.2.24.1.1.2
- loginFailureCountTableEntry - { loginFailureCountTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.1.2.1

Table 276. loginFailureCountTable

Object	OID	Syntax	Access	Status	Supported
loginFailureAccessApp	{ loginFailureCountTableEntry 1 }	INTEGER console(1) telnet(2) ftp(3)	read-only	current	N
loginFailureCount	{ loginFailureCountTableEntry 2 }	Counter32	read-only	current	N

5.29.4.1.2.1 loginFailureAccessApp

This object describes the access application used by the end user to access the device. This can be done through console, using telnet or using ftp. The following values are defined for this object:

- console(1)
- telnet(2)
- ftp(3)

5.29.4.1.2.2 loginFailureCount

This object contains the number of unsuccessful logins for console, ftp or telnet.

5.29.4.2 sysDevConfig

The configuration group. The implementation of this group is optional.

OIDs:

- sysDevConfig - { sysDevDslamMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.24.1.2

Table 277. sysDevConfig

Object	OID	Syntax	Access	Status	Supported
enablePowerSourceFailureAlarm	{ sysDevConfig 1 }	INTEGER disable(1) enable(2)	read-write	current	N
devIfTable	{ sysDevConfig 2 }	SEQUENCE OF devIfTableEntry	not-accessible	current	N
communityTrapAddressInfoTable	{ sysDevConfig 3 }	SEQUENCE OF communityTrapAddressInfoTableEntry	not-accessible	current	N
entCommunityTable	{ sysDevConfig 4 }	SEQUENCE OF entCommunityTableEntry	not-accessible	current	Y
sysDevUserAccountTable	{ sysDevConfig 5 }	SEQUENCE OF sysDevUserAccountEntry	not-accessible	current	N
sysDevIDSLConfigTable	{ sysDevConfig 6 }	SEQUENCE OF sysDevIdslConfigEntry	not-accessible	current	N
sysDevDslamSyslog	{ sysDevConfig 7 }	OID	N/A	N/A	N
sysDevConfigUserAccountTable	{ sysDevConfig 8 }	SEQUENCE OF sysDevConfigUserAccounntEntry	not-accessible	current	Y
sysDevConfigUserAccountIndexNext	{ sysDevConfig 9 }	Integer32	read-only	current	Y

5.29.4.2.1 enablePowerSourceFailureAlarm

This objects corresponds to enabling/disabling the power source failure alarm - for both A and B power sources This object is for the MCC only. The following values are defined for this object:

- disable (1)
- enable (2)

The default value of this object should be 1.

5.29.4.2.2 devIfTable

This table is used to configure information for a particular interface. This table is a sequence of defIfTableEntry where an entry is based on the ifindex.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- devIfTable - { sysDevConfig 2 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.2
- devIfTableEntry - { devIfTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.2.1

Table 278. devIfTable

Object	OID	Syntax	Access	Status	Supported
devPacketDiscardPolicy	{ devIfTableEntry 1 }	INTEGER noOp(1) mrrp(2) lrrp(3)	read-create	current	N
devLinkIntegrity	{ devIfTableEntry 2 }	INTEGER enable(1) disable(2) none(3)	read-create	current	N

5.29.4.2.2.1 devPacketDiscardPolicy

This object corresponds to the policy for packet discards during periods of congestion. The following values are defined for this object:

- noOp(1)
- mrrp(2) - most recently received packets are discarded
- lrrp(3) - least recently received packets are discarded

The default value of this object should be mrrp(2).

5.29.4.2.2.2 devLinkIntegrity

This object corresponds to the enabling or disabling of the Ethernet link integrity. The following values are defined for this object:

- enable(1) - enable link integrity
- disable(2) - disable link integrity
- none(3) - for interfaces that do not support link integrity

The default value of this object should be enable(1).

5.29.4.2.3 communityTrapAddressInfoTable

This table is used to set the trap destination address for a particular community. This table is a sequence of communityTrapAddressInofTableEntry.

Indices:

- trapCommunityName
- trapDestAndPort

OIDs:

- communityTrapAddressInfoTable - { sysDevConfig 3 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.3

- communityTrapAddressInfoTableEntry - { communityTrapAddressInfoTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.3.1

Table 279. communityTrapAddressInfoTable

Object	OID	Syntax	Access	Status	Supported
trapCommunityName	{ communityTrapAddressInfoTableEntry 1 }	DisplayString (SIZE(1…32))	not-accessible	current	N
trapDestAndPort	{ communityTrapAddressInfoTableEntry 2 }	TAddress	not-accessible	current	N
trapsEnable	{ communityTrapAddressInfoTableEntry 3 }	INTEGER enable (1) disable (2)	read-create	current	N
trapRowStatus	{ communityTrapAddressInfoTableEntry 4 }	RowStatus	read-only	current	N

5.29.4.2.3.1 trapCommunityName

This object corresponds to the name of the SNMP Community.

5.29.4.2.3.2 trapDestAndPort

The IP Address and Port of the destination to which a trap must be sent.

5.29.4.2.3.3 trapsEnable

This object corresponds to turning traps on/off for a particular destination. The following values are defined for this object:

- enable (1) - traps will be sent to the specified destination
- disable (2) - traps will not be sent to the specified destination

The default value of this object should be disable(2).

5.29.4.2.3.4 trapRowStatus

This object is used to add or delete a a row from the table.

5.29.4.2.4 entCommunityTable

This table is used to set the various configuration parameters for a particular community and is a sequence eof entCommunityTableEntry.

Index:

- entCommunityName IMPLIED

OIDs:

- entCommunityTable - { sysDevConfig 4 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.4
- entCommunityTableEntry - { entCommunityTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.4.1

Table 280. entCommunityTable

Object	OID	Syntax	Access	Status	Supported
entCommunityName	{ entCommunityTableEntry 1 }	DisplayString (SIZE(1…32))	not-accessible	current	Y
entCommunityType	{ entCommunityTableEntry 2 }	INTEGER readOnly(1) readWrite(2)	read-create	current	Y

Table 280. entCommunityTable

Object	OID	Syntax	Access	Status	Supported
entCommunityRowStatus	{ entCommunityTableEntry 3 }	RowStatus	read-create	current	Y

5.29.4.2.4.1 entCommunityName

This object corresponds to the name of the SNMP Community.

5.29.4.2.4.2 entCommunityType

This object defines the access allowed for a particular community. The following values are defined for this object:

- readOnly(1) - this community is only allowed to do gets
- readWrite(2) - this community is allowed to do gets and sets

The default value of this object should be readOnly(1).

5.29.4.2.4.3 entCommunityRowStatus

This object is used to add or delete a row from the table.

5.29.4.2.5 sysDevUserAccountTable

This table contains the user accounts. This table is a sequence of sysDevUserAccountEntry where an entry contains information relative to a particular user account.

Index:

- sysDevUserAccountUserId

OIDs:

- sysDevUserAccountTable - { sysDevConfig 5 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.5
- sysDevUserAccountEntry - { sysDevUserAccountTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.5.1

Table 281. sysDevUserAccountTable

Object	OID	Syntax	Access	Status	Supported
sysDevUserAccountUserId	{ sysDevUserAccountEntry 1 }	DisplayString (SIZE(6…15))	not-accessible	current	N
sysDevUserAccountPrivilege	{ sysDevUserAccountEntry 2 }	INTEGER operator(1) administrator(2)	read-create	current	N
sysDevUserAccountUserPassword	{ sysDevUserAccountEntry 3 }	DisplayString (SIZE(6…15))	read-only	current	N
sysDevUserAccountAccessPartition	{ sysDevUserAccountEntry 4 }	DisplayString (SIZE(0…50))	read-create	current	N
sysDevUserAccountRowStatus	{ sysDevUserAccountEntry 5 }	RowStatus	read-create	current	N

5.29.4.2.5.1 sysDevUserAccountUserId

This object corresponds to the login ID of the user account.

5.29.4.2.5.2 sysDevUserAccountPrivilege

This object corresponds to the access privilege of the user account. The following values are defined for this object:

- operator (1)

- administrator (2)

5.29.4.2.5.3 sysDevUserAccountUserPassword

This object corresponds to the password of the user account.

5.29.4.2.5.4 sysDevUserAccountAccessPartition

This object corresponds to the access partition of the user account. The default value of this object is “all”

5.29.4.2.5.5 sysDevUserAccountRowStatus

This object corresponds to create or delete a row in sysDevUserAccountTable.

5.29.4.2.6 sysDevIDSLConfigTable

A table that contains configuration information about IDSL Card. This table is a sequence of sysDevIDSLConfigEntry where an entry is a A list of information for IDSL device Configuration.

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- sysDevIDSLConfigTable - { sysDevConfig 6 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.6
- sysDevIDSLConfigEntry - { sysDevIDSLConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.6.1

Table 282. sysDevIDSLConfigTable

Object	OID	Syntax	Access	Status	Supported
sysDevIDSLConfigPrimaryNetClockMode	{ sysDevIDSLConfigEntry 1 }	IdslClockMode	read-create	current	N
sysDevIDSLConfigSecondaryNetClockMode	{ sysDevIDSLConfigEntry 2 }	IdslClockMode	read-create	current	N

5.29.4.2.6.1 sysDevIDSLConfigPrimaryNetClockMode

This object indicates the network clock mode set for the primary network clock. IDSL port cards with a port configured as an NT will be set to portCardDriveClockOnboard(4), both driving the backplane and using the clock for the other Local Timing transceivers on the card. Port cards with only LT ports configured will receive a clock from the backplane using portCardSinkClock(2). portCardDriveClock(3) will drive the backplane alone. The default value of this object is triState(1).

The default value for this object is triState(1).

5.29.4.2.6.2 sysDevIDSLConfigSecondaryNetClockMode

This object indicates the network clock mode set for the secondary network clock. IDSL port cards with a port configured as an NT will be set to portCardDriveClockOnboard(4), both driving the backplane and using the clock for the other Local Timing transceivers on the card. Port cards with only LT ports configured will receive a clock from the backplane using portCardSinkClock(2). portCardDriveClock(3) will drive the backplane alone. The default value of this object is triState(1).

The default value for this object is triState(1).

5.29.4.2.7 sysDevDslamSyslog

The syslog group.

OIDs:

- sysDevDslamSyslog - { sysDevConfig 7 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.7

Table 283. sysDevDslamSyslog

Object	OID	Syntax	Access	Status	Supported
sysDevSyslogFtpServerXferStatsEnable	{ sysDevDslamSyslog 1 }	SwitchState	read-write	current	N
sysDevSyslogTftpServerXferStatsEnable	{ sysDevDslamSyslog 2 }	SwitchState	read-write	current	N

5.29.4.2.7.1 sysDevSyslogFtpServerXferStatsEnable

This object allows the network manager to enable and disable syslog messages for FTP server file transfer statistics.

The default value is disabled(2).

5.29.4.2.7.2 sysDevSyslogTftpServerXferStatsEnable

This object allows the network manager to enable and disable syslog messages for TFTP server file transfer statistics.

The default value is disabled(2).

5.29.4.2.8 sysDevConfigUserAccountTable

This table contains the user accounts. This table is a sequence of sysDevConfigUserAccountEntry where an entry contains user account information.

Index:

- sysDevConfigUserAccountIndex

OIDs:

- sysDevConfigUserAccountTable - { sysDevConfig 8 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.8
- sysDevConfigUserAccountEntry - { sysDevConfigUserAccountTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.1.2.8.1

Table 284. sysDevConfigUserAccountTable

Object	OID	Syntax	Access	Status	Supported
sysDevConfigUserAccountIndex	{ sysDevConfigUserAccountEntry 1 }	Integer32	not-accessible	current	Y
sysDevConfigUserAccountUserId	{ sysDevConfigUserAccountEntry 2 }	DisplayString (SIZE(1…15))	read-create	current	Y
sysDevConfigUserAccountPrivilegedPassword	{ sysDevConfigUserAccountEntry 3 }	DisplayString (SIZE(0…15))	read-create	current	Y
sysDevConfigUserAccountUserPassword	{ sysDevConfigUserAccountEntry 4 }	DisplayString (SIZE(0…15))	read-create	current	Y
sysDevConfigUserAccountRowStatus	{ sysDevConfigUserAccountEntry 5 }	RowStatus	read-create	current	Y

5.29.4.2.8.1 sysDevConfigUserAccountIndex

The value of this object is an index of the account config table. This object is used by sysDevConfigUserAccountTable to identify the row in the table. When creating new row of this table the value of this object may be obtained by retrieving the value of sysDevConfigUserAccountIndexNext.

5.29.4.2.8.2 sysDevConfigUserAccountUserId

This object corresponds to the login ID of the user account.

5.29.4.2.8.3 sysDevConfigUserAccountPrivilegedPassword

This object corresponds to the access privilege password of the user account. If user is configured with this password in addition to a UserPassword than the user is Administrator. In a case that user is Operator and does not supply this password upon user configuration this value should be null string

5.29.4.2.8.4 sysDevConfigUserAccountUserPassword

This object corresponds to the password of the user account.

5.29.4.2.8.5 sysDevConfigUserAccountRowStatus

This object corresponds to create or delete a row in sysDevConfigUserAccountTable.

5.29.4.2.9 sysDevConfigUserAccountIndexNext

This object contains an appropriate value to be used for sysDevConfigUserAccountTable. The value of 0 indicates that no unassigned entries are available. To obtain sysDevConfigUserAccountIndex value for the new entry the manager issues a management protocol retrieval operation to obtain a current value of this object. After each retrieval, the agent should modify the value to the next unassigned index.

5.29.5 Traps

OIDs:

- sysDevDslamMIBTraps - { pdn-dslam 2 } 1.3.6.1.4.1.1795.2.24.2.24.2

Table 285. PDN-DSLAM-SYSTEM-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
cCN	{ sysDevDslamMIBTraps 7 }	ifIndex	current	N
authenticationFailureTrap	{ sysDevDslamMIBTraps 8 }	loginFailureAccessApp loginFailureCount	current	N
fanModuleFailure	{ sysDevDslamMIBTraps 9 }	*None*	current	N
fanModuleOperational	{ sysDevDslamMIBTraps 109 }	*None*	current	N
powerSourceAFailure	{ sysDevDslamMIBTraps 10 }	*None*	current	N
powerSourceAOperational	{ sysDevDslamMIBTraps 110 }	*None*	current	N
slotPollFailure	{ sysDevDslamMIBTraps 11 }	entPhysicalIndex	current	N
newCardDetected	{ sysDevDslamMIBTraps 111 }	entPhysicalIndex	current	N
ethernetJabber	{ sysDevDslamMIBTraps 12 }	ifIndex	current	N
ethernetJabberClear	{ sysDevDslamMIBTraps 112 }	ifIndex	current	N
ethernetJumbos	{ sysDevDslamMIBTraps 13 }	ifIndex	current	N
ethernetRunts	{ sysDevDslamMIBTraps 14 }	ifIndex	current	N
powerSourceBFailure	{ sysDevDslamMIBTraps 17 }	*None*	current	N
powerSourceBOperational	{ sysDevDslamMIBTraps 117 }	*None*	current	N
nonIpConservativeCardDetected	{ sysDevDslamMIBTraps 18 }	entPhysicalIndex	current	N
nonSupportedMCC	{ sysDevDslamMIBTraps 20 }	sysObjectID	current	N
nonSupportedChassis	{ sysDevDslamMIBTraps 21 }	sysObjectID	current	N
fanEntityModuleFailure	{ sysDevDslamMIBTraps 22 }	entPhysicalIndex	current	N
fanEntityModuleOperational	{ sysDevDslamMIBTraps 122 }	entPhysicalIndex	current	N

5.29.5.1 cCN

This trap signifies a Configuration change or software upgrade in the xDSL card. This trap is of “warning” class

5.29.5.2 authenticationFailureTrap

This trap signifies an authentication failure. Authentication failures can be telnet based or terminal based. This trap is in addition to the SNMP based authentication failure trap, which is a generic trap.

5.29.5.3 fanModuleFailure

This trap indicates the indicates the failure of the fan module on the device.

5.29.5.4 fanModuleOperational

This trap indicates the indicates the fan module on the device is operational.

5.29.5.5 powerSourceAFailure

This trap indicates that power source A has failed. This is sent only by the MCC card. This trap is of “minor” class.

5.29.5.6 powerSourceAOperational

This trap indicates that the power source A is operational. This is sent only by the MCC card. This trap is of “minor” class. This trap compliments powerSourceAFailure trap.

5.29.5.7 slotPollFailure

This trap indicates a slot poll failure. This is sent only by the MCC card. this trap is of “major” class.

5.29.5.8 newCardDetected

This trap indicates that a new card was detected in a slot. This is sent only by the MCC card. this trap is of “warning” class.

5.29.5.9 ethernetJabber

This trap indicates that a jabber condition has been detected on the ethernet interface. This trap is of “major” class.

5.29.5.10 ethernetJabberClear

This trap indicates that the jabber condition that was detected no longer exists. This trap is of “major” class.

5.29.5.11 ethernetJumbos

This trap indicates Ethernet jumbos. this trap is of “minor” class.

5.29.5.12 ethernetRunts

This trap indicates Ethernet runts. this trap is of “minor.”

5.29.5.13 powerSourceBFailure

This trap indicates that power source B has failed. This is sent only by the MCC card. This trap is of “minor” class.

5.29.5.14 powerSourceBOperational

This trap indicates that the power source B is operational. This is sent only by the MCC card. This trap is of “minor” class. This trap compliments powerSourceBFailure trap.

5.29.5.15 nonIpConservativeCardDetected

This trap indicates that a non ip conservative DSL card exist in the chassis at the slot “slotNumber.” This is sent only by the MCC card. This trap is of warning class.

5.29.5.16 nonSupportedMCC

AN has detected MCC firmware release too low to support this device.

5.29.5.17 nonSupportedChassis

AN in slot xx has been installed in a chassis that cannot support one or more of its features.

5.29.5.18 fanEntityModuleFailure

This trap indicates the indicates the failure of the fan module on the device.

5.29.5.19 fanEntityModuleOperational

This trap indicates the indicates the fan module on the device is operational.

5.30 PDN-ENTITY-SENSOR-EXT-MIB — PDN-ENTITY-SENSOR-EXT-MIB.mib

5.30.1 Introduction

This MIB module is a supplement to the ENTITY-SENSOR-MIB, RFC 3433.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200306060000Z,” or June 6, 2003.

5.30.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.30.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnEntitySensorExtMIB - { pdn-common 45 } 1.3.6.1.4.1.1795.2.24.2.45
- pdnEntitySensorExtNotifications - { pdnEntitySensorExtMIB 0 } 1.3.6.1.4.1.1795.2.24.2.45.0
- pdnEntitySensorExtObjects - { pdnEntitySensorExtMIB 1 } 1.3.6.1.4.1.1795.2.24.2.45.1
- pdnEntitySensorExtAFNs - { pdnEntitySensorExtMIB 2 } 1.3.6.1.4.1.1795.2.24.2.45.2

5.30.4 MIB Objects

Table 286. PDN-ENTITY-SENSOR-EXT-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnEntPhySensorExtTable	{ pdnEntitySensorExtObjects 1 }	SEQUENCE OF pdnEntPhySensorExtEntry	not-accessible	current	Y

5.30.4.1 pdnEntPhySensorExtTable

This table extends the entPhySensorTable.

Augments:

- entPhySensorTable

Index:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- pdnEntPhySensorExtTable - { pdnEntitySensorExtObjects 1 } 1.3.6.1.4.1.1795.2.24.2.45.1.1
- pdnEntPhySensorExtEntry - { pdnEntPhySensorExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.45.1.1.1

Table 287. pdnEntPhySensorExtTable

Object	OID	Syntax	Access	Status	Supported
pdnEntPhySensorExt NotificationEnable	{ pdnEntPhySensorExtEntry 1 }	BITS thresholdExceeded(0)	read-write	current	Y
pdnEntPhySensorExt UpperThreshold	{ pdnEntPhySensorExtEntry 2 }	EntitySensorValue	read-write[a]	current	Y
pdnEntPhySensorExt LowerThreshold	{ pdnEntPhySensorExtEntry 3 }	EntitySensorValue	read-write[a]	current	Y
pdnEntPhySensorExt ThresholdState	{ pdnEntPhySensorExtEntry 4 }	INTEGER noThresholdsExceeded(1) upperThresholdExceeded(2) lowerThresholdExceeded(3)	read-only	current	Y

a. Thresholds for the Intake Temperature Sensor are supported with an access of read-write; all other sensors are supported with an access of read-only.

5.30.4.1.1 pdnEntPhySensorExtNotificationEnable

Provides that ability to enable and disable notifications relative to objects in this table. The following are valid values:

- thresholdExceeded(0) - When this bit is set, pdnEntPhySensorExtThresholdExceededSet and pdnEntPhySensorExtThresholdExceededCleared notifications should be generated. When this bit is reset, pdnEntPhySensorExtThresholdExceededSet, and pdnEntPhySensorExtThresholdExceededCleared notifications should be not be generated.

5.30.4.1.2 pdnEntPhySensorExtUpperThreshold

This object sets the upper limit of a sensor's threshold. When the value of entPhySensorValue becomes greater than the value of this object, an "Upper Threshold Exceeded" state is entered.

Valid Object Value(s)

The range of valid values for the Intake Temperature Sensor is -40…65.

There is no upper limit for the Fan Speed Sensor and a read/get will return 999 999 999 (without spaces).

5.30.4.1.3 pdnEntPhySensorExtLowerThreshold

This object sets the lower limit of a sensor's threshold. When the value of entPhySensorValue becomes less than the value of this object, a "Lower Threshold Exceeded" state is entered.

Valid Object Value(s)

The range of valid values for the Intake Temperature Sensor is -40…65.

5.30.4.1.4 pdnEntPhySensorExtThresholdState

This object returns the threshold state of the sensor. The following are valid values:

- noThresholdsExceeded(1)
- upperThresholdExceeded(2)
- lowerThresholdExceeded(3)

5.30.5 Traps

This section defines the traps generated.

OIDs

- pdnEntitySensorExtNotifications - { pdnEntitySensorExtMIB 0 } 1.3.6.1.4.1.1795.2.24.2.45.0

Table 288. PDN-ENTITY-SENSOR-EXT-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
pdnEntPhySensorExt ThresholdExceededSet	{ pdnEntitySensorExtNotifications 1 }	entPhySensorValue pdnEntPhySensorExtThresholdState	current	Y
pdnEntPhySensorExt Threshold ExceededCleared	{ pdnEntitySensorExtNotifications 100 }	entPhySensorValue pdnEntPhySensorExtThresholdState	current	Y

5.30.5.1 pdnEntPhySensorExtThresholdExceededSet

This trap/notification signifies that a sensor value has exceeded its threshold limit: that is, its entPhySensorValue is greater than its pdnEntPhySensorExtUpperThreshold or less than its pdnEntPhySensorExtLowerThreshold.

5.30.5.2 pdnEntPhySensorExtThresholdExceededCleared

This trap/notification signifies that a sensor value that had exceeded its threshold limit, is now operating with in its threshold limits: that is, its entPhySensorValue is less than or equal to its pdnEntPhySensorExtUpperThreshold and greater than or equal to its pdnEntPhySensorExtLowerThreshold.

5.30.6 Conformance Information

5.30.6.1 Compliance

5.30.6.1.1 pdnEntitySensorExtMIBCompliance

The compliance statement for pdnEntitySensorExt entities which implement the pdnEntitySensorExtMIB.

Mandatory Groups:

None specified.

Optional Groups:

- pdnEntitySensorExtThresholdGroup - This group is mandatory only for those entities which implement setting and reporting of thresholds.
- pdnEntitySensorExtThresholdNtfyGroup - This group is mandatory only for those entities which implement traps for thresholds being exceeded.

Objects that have a MIN-ACCESS defined:

- pdnEntPhySensorExtNotificationEnable - MIN-ACCESS read-only
  In some implementations, for some sensors, it may not be allowable to enable/disable the notification.
- pdnEntPhySensorExtUpperThreshold - MIN-ACCESS read-only
  In some implementations, sensor thresholds may not be configurable by the user. Specific hardware implementations may dictate thresholds. In these cases, the object can be supported as read-only access.
- pdnEntPhySensorExtLowerThreshold - MIN-ACCESS read-only
  In some implementations, sensor thresholds may not be configurable by the user. Specific hardware implementations may dictate thresholds. In these cases, the object can be supported as read-only access.

5.30.6.2 Conformance

Table 289. PDN-ENT-SENSOR-EXT-MIB Conformance

Group	Description	Objects	Supported
pdnEntitySensorExtThresholdGroup	A collection of objects for setting and reporting thresholds.	pdnEntPhySensorExtNotificationEnable	Y
		pdnEntPhySensorExtUpperThreshold	Y
		pdnEntPhySensorExtLowerThreshold	Y
		pdnEntPhySensorExtThresholdState	Y
		pdnEntPhySensorExtIndex	Y
pdnEntitySensorExtThresholdNtfyAfnGroup	Accessible for notify objects to support notifications that thresholds were exceeded.	pdnEntPhySensorExtIndex	Y

Table 289. PDN-ENT-SENSOR-EXT-MIB Conformance

Group	Description	Objects	Supported
pdnEntitySensorExtThresholdNtfyGroup	Notifications relative to thresholds going out of or in to specification.	pdnEntPhySensorExtThresholdExceededSet	Y
		pdnEntPhySensorExtThresholdExceededCleared	Y

5.31 PDN-ETHER-MIB — pdn_ether.mib

5.31.1 Introduction

This MIB contains objects that are used to configure Ethernet interfaces.

The version of the MIB used for this document had a LAST-UPDATED of 0205100000Z which decodes to February 10, 2002.

5.31.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- ManagementType - This object is used to specify the type of management. This can be either Inband Management or OutBand Management.
  — SYNTAX INTEGER
  - inband(1)
  - outband(2)
- ResetStates - This object defines the enumerations of values that can be applied to the reset object.
  —SYNTAX INTEGER
  - noOp(1)
  - reset(2)
  - resetToFactoryDefaults(3)

5.31.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-ether - { pdn-common 18 } 1.3.6.1.4.1.1795.2.24.2.18
- pdnPortConfigMIBOBjects - { pdn-ether 1 } 1.3.6.1.4.1.1795.2.24.2.18.1
- pdnPortConfigMIBTraps - { pdn-ether 2 } 1.3.6.1.4.1.1795.2.24.2.18.2
- pdnPortConfigEthernet - { pdnPortConfigMIBOBjects 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.1
- pdnPortConfigGroups - { pdnPortConfigMIBOBjects 2 } 1.3.6.1.4.1.1795.2.24.2.18.1.2
- pdnPortConfigMauExtMIBObject - { pdnPortConfigMIBOBjects 3 } 1.3.6.1.4.1.1795.2.24.2.18.1.3
- pdnPortConfigIfJackMIBObject - { pdnPortConfigMIBOBjects 4 } 1.3.6.1.4.1.1795.2.24.2.18.1.4

5.31.4 MIB Objects

Table 290. PDN-ETHER-MIB Objects

Object	OID	Description	Supported
pdnPortConfigEthernet	{ pdnPortConfigMIBObjects 1 }	Ethernet Port Configuration Group	N

Table 290. PDN-ETHER-MIB Objects

Object	OID	Description	Supported
pdnPortConfiigMauExtMIBObjects	{ pdnPortConfigMIBObjects 3 }	MAU Configuration Group	Y
pdnPortConfigIfJackMIBObject	{ pdnPortConfigMIBObjects 4 }	Jack Configuration Group	Y

5.31.4.1 pdnPortConfigEthernet

The Ethernet port configuration group.

Table 291. pdnPortConfigEthernet

Object	OID	Syntax	Access	Status	Supported
pdnPortConfigEthernetTable	{ pdnPortConfigEthernet 1 }	SEQUENCE OF pdnPortConfigEthernetEntry	not-accessible	current	N

5.31.4.1.1 pdnPortConfigEthernetTable

A Table that is a sequence of pdnPortConfigEthernetEntry where each entry contains information about the Ethernet port configuration.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- pdnPortConfigEthernetTable - { pdnPortConfigEthernet 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.1.1
- pdnPortConfigEthernetEntry - { pdnPortConfigEthernetTable 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.1.1.1

Table 292. pdnPortConfigEthernetTable

Object	OID	Syntax	Access	Status	Supported
pdnPortConfigEthernetDuplexMode	{ pdnPortConfigEthernetEntry 1 }	SwitchState	read-create	current	N
loginTimpdnPortConfigEthernetManageType	{ pdnPortConfigEthernetEntry 2 }	ManagementType	read-create	current	N
pdnPortConfigEthernetResetState	{ pdnPortConfigEthernetEntry 3 }	ResetStates	read-create	current	N
pdnPortConfigEthernetAutoNegotiate	{ pdnPortConfigEthernetEntry 4 }	SwitchState	read-create	current	N
pdnPortConfigEthernetSpeed	{ pdnPortConfigEthernetEntry 5 }	INTEGER tenBaseT(1) hundered-BaseT(2)	read-create	current	N

5.31.4.1.1.1 pdnPortConfigEthernetDuplexMode

This object is used to enable or Disable Duplex Mode on an Ethernet Port. This object can not be enabled if the auto-negotiate mode is enabled. The following values are defined for this object:

- enabled(1)
- disabled(2)

The default value for this object is disabled(2).

5.31.4.1.1.2 pdnPortConfigEthernetManageType

This object is used to Enable or Disable Inband or Outband Management. The following values are defined for this object:

- inband(1)
- outband(2)

The default value for this object is outband(2).

5.31.4.1.1.3 pdnPortConfigEthernetResetState

This object is used to reset the Ethernet port. The following values are defined for this object:

- noOp(1)
- reset(2)
- resetToFactoryDefaults(3)

The default value for this object is noOp(1).

5.31.4.1.1.4 pdnPortConfigEthernetAutoNegotiate

This object is used to enable or disable the auto-negotiate mode. Enabling this mode will disable selection of the speed and/or full duplex mode. The following values are defined for this object:

- enabled(1)
- disabled(2)

The default value for this object is enabled(1).

5.31.4.1.1.5 pdnPortConfigEthernetSpeed

This object corresponds to the speed of the Ethernet port. This object can not be set if the auto-negotiate mode is enabled. The following values are defined for this object:

- tenBaseT(1) - 10BaseT
- hundredBaseT(2) - 100BaseT

The default value for this object is tenBaseT(1).

5.31.4.2 pdnPortConfigMauExtMIBObject

The MAU configuration group.

Table 293. pdnPortConfigMauExtMIBObject

Object	OID	Syntax	Access	Status	Supported
pdnPortConfigMxuExtTable	{ pdnPortConfigMauExtMIBObject 1 }	SEQUENCE OF pdnPortConfigMauExtEntry	not-accessible	current	Y

5.31.4.2.1 pdnPortConfigMauExtTable

A Table that contains configuration information about the Ethernet port configuration and it extends standard MAU table from RFC 2668. This table is a sequence of pdnPortConfigExtEntry where an entry contains additional information about an Ethernet port.

Augments:

- ifMauTable (Section 5.9.4.2.1, Page 141)

Indices:

- ifMauIfIndex (Section 5.9.4.2.1.1, Page 143)
- ifMauIndex (Section 5.9.4.2.1.2, Page 143)

OIDs:

- pdnPortConfigMauExtTable - { pdnPortConfigMauExtMIBObject 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.3.1
- pdnPortConfigMauExtEntry - { pdnPortConfigMauExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.3.1.1

Table 294. pdnPortConfigMauExtTable

Object	OID	Syntax	Access	Status	Supported
pdnPortConfigXover	{ pdnPortConfigMauExtEntry 1 }	INTEGER mdi(1) mdix(2)	read-create	current	Y

5.31.4.2.1.1 pdnPortConfigXover

This object is used to configure if crossover or straight-through cable is used on Ethernet interface if the interface is not set for autonegotiontion. The following values are defined for this object:

- mdi(1) - straight-through
- mdix(2) - crossover

5.31.4.3 pdnPortConfigIfJackMIBObject

The Ethernet jack configuration group.

Table 295. pdnPortConfigIfJackMIBObject

Object	OID	Syntax	Access	Status	Supported
pdnIfJackTable	{ pdnPortConfigIfJackMIBObject 1 }	SEQUENCE OF pdnIfJackEntry	not-accessible	current	Y

5.31.4.3.1 pdnIfJackTable

This table conceptually extends the ifJackTable but not as an augmentation. This table is a sequence of pdnIfJackEntry where each entry contains configuration information about a jack.

NOTE:

The original desire was to implement this table as an augmentation of the ifJackTable in the 802.3 MAU MIB specified in RFC 2668 and call it pdnIfJackExtTable. However, the SNMP Research MIB Tools would not successfully compile an augmentation of this table with an object that is read-write because the ifJackTable only has read-only objects.

Indices:

- ifMauIfIndex (Section 5.9.4.2.1.1, Page 143)
- ifMauIndex (Section 5.9.4.2.1.2, Page 143)
- ifJackIndex (Section 5.9.4.2.2.1, Page 150)

OIDs:

- pdnIfJackTable - { pdnPortConfigIfJackMIBObject 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.4.1

- pdnIfJackEntry - { pdnIfJackTable 1 } 1.3.6.1.4.1.1795.2.24.2.18.1.4.1.1

Table 296. pdnIfJackTable

Object	OID	Syntax	Access	Status	Supported
pdnActiveJack	{ pdnIfJackEntry 1 }	INTEGER fiber(1) rj45(2) auto(3)	read-write	current	Y

5.31.4.3.1.1 pdnActiveJack

The following values are defined for this object:

- fiber(1) - fiber interface (could be a Small Form-factor Pluggable (SFP) module)
- rj45(2) - Copper wire interface
- auto(3) - Automatically detects fiber(1) or rj45(2)

This object allows the jack type to be configured or a priority to be set. This applies to Ethernet interfaces that may support more than one type of jack on an interface and only one jack is valid at a time.

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM support two types of jacks, a RJ45 copper wire interface and s SFP fiber interface. The product is required to have the user select which jack has priority when both jacks are active.

Valid Object Value(s)

- fiber(1)-The fiber jack has priority.
- rj45(2)-The rj45 jack has priority.
- auto(3)-Not used.

5.32 PDN-FILTER-MIB — pdn_filter.mib

5.32.1 Introduction

This MIB contains objects that are used to configure filters.

The version of the MIB used for this document had a comment stating that it was last updated on January, 11, 2001.

5.32.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- VnidRange - The valid range for VNID IDs.
  — SYNTAX INTEGER(2…4000)

5.32.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-filter - { pdn-common 23 } 1.3.6.1.4.1.1795.2.24.2.23
- sysDevFilterMIBObjects - { pdn-filter 1 } 1.3.6.1.4.1.1795.2.24.2.23.1
- sysDevFilterMIBTraps - { pdn-filter 2 } 1.3.6.1.4.1.1795.2.24.2.23.2
- sysDevFilter - { sysDevFilterMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1
- sysDevIpFilter - { sysDevFilterMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.23.1.2

5.32.4 MIB Objects

Table 297. PDN-FILTER-MIB Objects

Object	OID	Description	Supported
sysDevFilter	{ sysDevFilterMIBObjects 1 }	System Filter Group	Y
sysDevIpFilter	{ sysDevIpFilter 2 }	IP Filter Group	N

5.32.4.1 sysDevFilter

The System Filter Group

OIDs:

- sysDevFilter - { sysDevFilterMIBObjects 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1

Table 298. sysDevFilter

Object	OID	Syntax	Access	Status	Supported
sysDevSNInjectionType	{ sysDevFilter 1 }	Integer ipFilter(1)	not-accessible	mandatory	N
sysDevSNInjectionVnid	{ sysDevFilter 2 }	VnidRange	not-accessible	mandatory	N
sysDevFilterConfigTable	{ sysDevFilter 3 }	SEQUENCE OF sysDevFilterConfigTableEntry	not-accessible	mandatory	Y
sysDevL2FilterRuleConfigTable	{ sysDevFilter 4 }	SEQUENCE OF sysDevL2FilterRuleConfigTableEntry	not-accessible	mandatory	Y
sysDefFilterBindingTable	{ sysDevFilter 5 }	SEQUENCE OF sysDefFilterBindingTableEntry	not-accessible	mandatory	Y
sysDevFilterIndexNext	{ sysDevFilter 6 }	INTEGER	read-only	mandatory	Y
sysDevL2FilterRuleIndexNext	{ sysDevFilter 7 }	INTEGER	read-only	mandatory	Y
sysDevFilterToRuleBindingTable	{ sysDevFilter 8 }	SEQUENCE OF sysDevFilterToRuleBindingTableEntry	not-accessible	mandatory	Y
sysDevL3FilterRuleConfigTable	{ sysDevFilter 9 }	SEQUENCE OF sysDevL3FilterRuleConfigTableEntry	not-accessible	mandatory	N
sydDevL3FilterRuleIndexNext	{ sysDevFilter 10 }	INTEGER	read-only	mandatory	N

5.32.4.1.1 sysDevSNInjectionType

The value of this object corresponds to the remote injection type.

5.32.4.1.2 sysDevSNInjectionVnid

This object corresponds to the vnid id (bound to an interface) on which filter injection is not supported or has failed.

5.32.4.1.3 sysDevFilterConfigTable

This table allows the user to configure filtering information. Every filter is comprised of a number of rules (see SysDevRuleTable). When a filter is bound to an interface, all the rules of that particular filter apply to that particular interface. Also the same filter maybe bound to several interfaces. This table only allows us to configure information about a particular filter. this table is indexed by sysDevFilterName.

The sysDevFilterConfigTable is used to create new filters. To create new filter, filter index, filter name, type and default action are necessary to be specified. Next the user should create rules that will be attached to his filter. To create rule user should use L2 or L3, L4 FilterRuleTable) depending on what type of rules user needs to create. To create L2 (layer two) rule user needs to specify rule index, rule name, ether type (RFC 1700) and frame type that will be accepted. The user should also configure what action should be taken if the rule is matched to a frame. The ether type can be a range of hex numbers (2bytes) so range objects should be used. Next the rule should be bound to one or more filters that are created. This binding is done by using FilterToRule binding table. This table is indexed by rule index and filter index. Next the filter-rule binding (completed filter) should be bound to an interface by using FilterBindingTable. This table is indexed by ifIndex (Section 5.3.4.3.1, Page 37) (interface we want filter bound to), filter index and direction that filter/rule combination is applied to (inbound, outbound and inboundoutbound). After this step the filter will take in effect and every packet on this interface will be examined for rule or set of rules that are bound to this filter. Filter should not be bound to an interface if the rule is not bound to a filter. it can be done but the default action will take in effect and every packet will be checked. The rule cannot be deleted until all filter-rule bindings for this filter are deleted. The rule cannot be deleted until all filter-interface bindings for this filter are disabled.

Index:

- sysDevFilterIndex

OIDs:

- sysDevFilterConfigTable - { sysDevFilter 3 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.3
- sysDevFilterConfigTableEntry - { sysDevFilterConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.3.1

Table 299. sysDevFilterConfigTable

Object	OID	Syntax	Access	Status	Supported
sysDevFilterIndex	{ sysDevFilterConfigTableEntry 1 }	INTEGER	read-only	mandatory	Y
sysDevFilterName	{ sysDevFilterConfigTableEntry 2 }	DisplayString (SIZE(1…12))	read-write	mandatory	Y
sysDevFilterType	{ sysDevFilterConfigTableEntry 3 }	INTEGER layer1(1) layer2(2) layer3(3) layer4(4) layer5(5) layer6(6) layer7(7) unknown(8)	read-write	mandatory	Y
sysDevDefFilterAction	{ sysDevFilterConfigTableEntry 4 }	INTEGER forward(1) discard(2)	read-write	mandatory	Y
sysDevFilterNumOfDynamicRules	{ sysDevFilterConfigTableEntry 5 }	INTEGER	read-only	mandatory	Y
sysDevFilterNumOfStaticRules	{ sysDevFilterConfigTableEntry 6 }	INTEGER	read-only	mandatory	Y
sysDevFilterRefCount	{ sysDevFilterConfigTableEntry 7 }	INTEGER	read-only	mandatory	Y
sysDevFilterRowStatus	{ sysDevFilterConfigTableEntry 8 }	RowStatus	read-write	mandatory	Y

5.32.4.1.3.1 sysDevFilterIndex

The value of this object is an index of the filter. This object is used by sysDevFilterConfigTable to identify the row in the table. When creating new row of this table the value of this object may be obtained by retrieving the value of sysDevFilterIndexNext, Section 5.32.4.1.6, Page 427.

5.32.4.1.3.2 sysDevFilterName

The value of this object corresponds directly to the name of the filter.

5.32.4.1.3.3 sysDevFilterType

The value of this object corresponds directly to the type of the filter. The following values are defined for this object:

- layer1(1)
- layer2(2)
- layer3(3)
- layer4(4)
- layer5(5)
- layer6(6)
- layer7(7)

- unknown(8)

Ethernet is layer 2 filter, Ip are layer 3 filters (include layer 4 protocol filtering).

5.32.4.1.3.4 sysDevDefFilterAction

The value of this object corresponds to the default action taken on a particular IP packet based on the filter rule. The following values are defined for this object:

- forward - the packet is forwarded
- discard - the packet is discarded

5.32.4.1.3.5 sysDevFilterNumOfDynamicRules

This object corresponds to the number of dynamic rules configured for this filter.

Default is zero.

5.32.4.1.3.6 sysDevFilterNumOfStaticRules

This object corresponds to the number of static rules configured for this filter.

Default is zero.

5.32.4.1.3.7 sysDevFilterRefCount

This object corresponds to the number of references for this filter by the various interfaces of the system For Example if filter “abc” is bound to two of the DSL interfaces as inbound filter then the reference count for this filter is 2. Filters can be deleted from the system (by a user) only if the reference count equals 0. (the filter is not bound to any interfaces).

The default is 0.

5.32.4.1.3.8 sysDevFilterRowStatus

The object is used to add or delete a row from this table.

5.32.4.1.4 sysDevL2FilterRuleConfigTable

This table allows the user to configure the rules of each IP filter. When a filter is bound to an interface, all the rules of that particular filter apply to that particular interface. Also, the same filter may be found to several interfaces.

Index:

- sysDevL2FilterRuleIndexNext (Section 5.32.4.1.7, Page 427)

OIDs:

- sysDevL2FilterRuleConfigTable - { sysDevFilter 4 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.4
- sysDevL2FilterRuleConfigTableEntry - { sysDevL2FilterRuleConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.4.1

Table 300. sysDevL2FilterRuleConfigTable

Object	OID	Syntax	Access	Status	Supported
sysDevL2FilterRuleIndex	{ sysDevL2FilterRuleConfigTableEntry 1 }	INTEGER	read-only	mandatory	Y
sysDevL2FilterRuleName	{ sysDevL2FilterRuleConfigTableEntry 2 }	DisplayString (SIZE(1…12))	read-write	mandatory	Y
sysDevL2FilterRuleEtherFrameType	{ sysDevL2FilterRuleConfigTableEntry 3 }	INTEGER dix(1) snap(2)	read-write	mandatory	Y
sysDevL2FilterRuleEtherType	{ sysDevL2FilterRuleConfigTableEntry 4 }	INTEGER range(1) singleType(2)	read-write	mandatory	Y

Table 300. sysDevL2FilterRuleConfigTable

Object	OID	Syntax	Access	Status	Supported
sysDevL2FilterRuleEtherTypeRangeStarts	{ sysDevL2FilterRuleConfigTableEntry 5 }	INTEGER	read-write	mandatory	Y
sysDevL2FilterRuleEtherTypeRangeEnds	{ sysDevL2FilterRuleConfigTableEntry 6 }	INTEGER	read-write	mandatory	N
sysDevL2FilterRuleAction	{ sysDevL2FilterRuleConfigTableEntry 7 }	INTEGER forward(1) discard(2)	read-write	mandatory	Y
sysDevL2FilterRuleRowStatus	{ sysDevL2FilterRuleConfigTableEntry 8 }	RowStatus	read-write	mandatory	Y

5.32.4.1.4.1 sysDevL2FilterRuleIndex

The value of this object is an index of the filter rule table. This object is used by sysDevFilterConfigTable to identify the row in the table. When creating new row of this table the value of this object may be obtained by retrieving the value of sysDevL2FilterRuleIndexNext, (Section 5.32.4.1.7, Page 427).

5.32.4.1.4.2 sysDevL2FilterRuleName

The value of this object corresponds directly to the name of the rule.

5.32.4.1.4.3 sysDevL2FilterRuleEtherFrameType

This value describes if Ethernet frame types defined that can be transmitted. The following values are defined for this object:

- dix(1)
- snap(2)

5.32.4.1.4.4 sysDevL2FilterRuleEtherType

This value describes if Ethernet types defined in RFC 1700 are specified as range or one single type. It is possible to create rule using a range of types and therefore next object is added to create start of the range. This object is the first in range and the value of it has to be smaller or equal to a hex Ethernet type value entered in sysDevL2FilterRuleEtherTypeRangeEnds. If this object specifies that the value used is a single type than sysDevL2FilterRuleEtherTypeRangeEnds will not be considered valid. The following values are defined for this object:

- range(1)
- singleType(2)

Valid Object Value(s)

- singleType(2)

5.32.4.1.4.5 sysDevL2FilterRuleEtherTypeRangeStarts

This value corresponds to Ethernet types defined in RFC 1700. It is possible to create rule using a range of types and therefore next object is added to create range. This object is the first in range and the value of it has to be smaller or equal to a hex Ethernet type value entered next. If the value of sysDevL2FilterRuleEtherType is singleType(2) then only this value is considered.

5.32.4.1.4.6 sysDevL2FilterRuleEtherTypeRangeEnds

This value corresponds to Ethernet types defined in RFC 1700. It is possible to create rule using a range of types and therefore this object is added to create range. This object is the second value in the range. The value of it has to be larger or equal to a hex Ethernet type value entered previously. This object is not considered valid if the value of sysDevL2FilterRuleEtherType is singleType(2).

5.32.4.1.4.7 sysDevL2FilterRuleAction

The value of this object specify the action to be done if the rule is matched to a packet. The following values are defined for this object:

- forward (1)
- discard (2)

5.32.4.1.4.8 sysDevL2FilterRuleRowStatus

The object is used to add or delete a row from this table.

5.32.4.1.5 sysDevFilterBindingTable

This table allows us to bind filters to a particular interface. As such this table is indexed by ifIndex, Section 5.3.4.3.1, Page 37, and by sysDevFilterBindingIndex.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- sysDevFilterBindingIndex
- sysDevFilterBindingDirection

OIDs:

- sysDevFilterBindingTable - { sysDevFilter 5 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.5
- sysDevFilterBindingTableEntry - { sysDevFilterBindingTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.5.1

Table 301. sysDevFilterBindingTable

Object	OID	Syntax	Access	Status	Supported
sysDevFilterBindingIndex	{ sysDevFilterBindingTableEntry 1 }	INTEGER	read-only	mandatory	Y
sysDevFilterBindingDirection	{ sysDevFilterBindingTableEntry 2 }	INTEGER inboundFilter(1) outboundFilter(2) inboundOutboundFilter(3)	read-only	mandatory	Y
sysDevFilterBindingAdminStatus	{ sysDevFilterBindingTableEntry 3 }	INTEGER up(1) down(2)	read-write	mandatory	Y
sysDevFilterBindingOperStatus	{ sysDevFilterBindingTableEntry 4 }	INTEGER up(1) down(2)	read-write	mandatory	Y
sysDevFilterBindingRowStatus	{ sysDevFilterBindingTableEntry 5 }	RowStatus	read-write	mandatory	Y

5.32.4.1.5.1 sysDevFilterBindingIndex

The value of this object corresponds directly to the filter index from sysDevFilterConfigTable.

5.32.4.1.5.2 sysDevFilterBindingDirection

The value of this object corresponds directly to the type of the filter which needs to be bound to a particular interface. The following values are defined for this object:

- inboundFilter - This indicates that the filter is applied to the packets coming in to the interface.
- outboundFilter - this indicates that the filter is applied to packets going out the interface.
- inboundOutboundFilter - this indicates that the filter is applied packets going out the interface and to packets coming in to the interface.

Valid Object Value(s)

- inboundFilter

5.32.4.1.5.3 sysDevFilterBindingAdminStatus

The value of this is used to configure admin status of filter binding. The following values are defined for this object:

- up (1) - this binding is active for the specified interface
- down (2) - this binding is inactive for specified interface

If the filter is bound to more than one interface the bindings on the other interfaces will have to be configured separately. To read this object use sysDevFilterBindingOperStatus.

5.32.4.1.5.4 sysDevFilterBindingOperStatus

The value of this is used to read oper status of filter binding. The following values are defined for this object:

- up (1) - the binding is active for specified interface
- down (2) - the binding is inactive for specified interface

To configure this object use sysDevFilterBindingAdminStatus.

5.32.4.1.5.5 sysDevFilterBindingRowStatus

This object is used to add or delete a row from this table.

5.32.4.1.6 sysDevFilterIndexNext

This object contains an appropriate value to be used for sysDevFilterConfigTable (Section 5.32.4.1.3, Page 422). The value of 0 indicates that no unassigned entries are available. To obtain sysDevFilterIndex value for the new entry the manager issues a management protocol retrieval operation to obtain a current value of this object. After each retrieval, the agent should modify the value to the next unassigned index.

5.32.4.1.7 sysDevL2FilterRuleIndexNext

This object contains an appropriate value to be used for sysDevL2FilterRuleConfigTable (Section 5.32.4.1.4, Page 424). The value of 0 indicates that no unassigned entries are available. To obtain sysDevL2FilterRuleIndex (Section 5.32.4.1.4.1, Page 425) value for the new entry the manager issues a management protocol retrieval operation to obtain a current value of this object. After each retrieval, the agent should modify the value to the next unassigned index.

5.32.4.1.8 sysDevFilterToRuleBindingTable

This table allows us to bind filters to a particular rules.

Indices:

- sysDevFilterIndex (Section 5.32.4.1.3.1, Page 423)
- sysDevFilterBindingIndex (Section 5.32.4.1.5.1, Page 426)

OIDs:

- sysDevFilterToRuleBindingTable - { sysDevFilter 8 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.8
- sysDevFilterToRuleBindingTa.8.1bleEntry - { sysDevFilterToRuleBindingTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1

Table 302. sysDevFilterToRuleBindingTable

Object	OID	Syntax	Access	Status	Supported
sysDevFilterToRuleBindingIndex	{ sysDevFilterToRuleBindingTableEntry 1 }	INTEGER	read-only	mandatory	Y
sysDevFilterToRulePriority	{ sysDevFilterToRuleBindingTableEntry 2 }	INTEGER	read-write	mandatory	Y
sysDevFilterToRuleBindingRowStatus	{ sysDevFilterToRuleBindingTableEntry 3 }	RowStatus	read-write	mandatory	Y

5.32.4.1.8.1 sysDevFilterToRuleBindingIndex

The value of this object corresponds directly to the rule index from sysDevL2FilterRuleConfigTable (Section 5.32.4.1.4, Page 424).

5.32.4.1.8.2 sysDevFilterToRulePriority

Internally the table is sorted based on this object- in the increasing order of priority. The table lookups are done on the sorted table.

5.32.4.1.8.3 sysDevFilterToRuleBindingRowStatus

This object is used to add or delete a row from this table.

5.32.4.1.9 sysDevL3FilterRuleConfigTable

This table allows the user to configure the rules for layer three filter. When a filter is bound to an interface, all the rules of that particular filter apply to the specified interface. Also, the same filter may be bound to several interfaces.

Index:

- sysDevL3FilterRuleIndex (Section 5.32.4.1.9.1, Page 429)

OIDs:

- sysDevL3FilterRuleConfigTable - { sysDevFilter 9 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.9
- sysDevL3FilterRuleConfigTableEntry - { sysDevL3FilterRuleConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.1.9.1

Table 303. sysDevL3FilterRuleConfigTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
sysDevL3FilterRuleIndex	{ sysDevL3FilterRuleConfigTableEntry 1 }	INTEGER	read-only	mandatory	N
sysDevL3FilterRuleName	{ sysDevL3FilterRuleConfigTableEntry 2 }	DisplayString	read-write	mandatory	N
sysDevL3FilterRuleSrcAddress	{ sysDevL3FilterRuleConfigTableEntry 3 }	IpAddress	read-write	mandatory	N
sysDevL3FilterRuleSrcAddrMask	{ sysDevL3FilterRuleConfigTableEntry 4 }	IpAddress	read-write	mandatory	N
sysDevL3FilterRuleSrcAddrAction	{ sysDevL3FilterRuleConfigTableEntry 5 }	INTEGER forward(1) discard(2) none(3)	read-write	mandatory	N
sysDevL3FilterRuleSrcPortNum	{ sysDevL3FilterRuleConfigTableEntry 6 }	INTEGER	read-write	mandatory	N
sysDevL3FilterRuleMaxSrcPortNum	{ sysDevL3FilterRuleConfigTableEntry 7 }	INTEGER	read-write	mandatory	N
sysDevL3FilterRuleSrcCompType	{ sysDevL3FilterRuleConfigTableEntry 8 }	INTEGER none(1) eq(2) neq(3) gt(4) lt(5) inRange(6) outRange(7)	read-write	mandatory	N
sysDevL3FilterRuleDestAddress	{ sysDevL3FilterRuleConfigTableEntry 9 }	IpAddress	read-write	mandatory	N
sysDevL3FilterRuleDestAddrMask	{ sysDevL3FilterRuleConfigTableEntry 10 }	IpAddress	read-write	mandatory	N

Table 303. sysDevL3FilterRuleConfigTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
sysDevL3FilterRuleDestAddrAction	{ sysDevL3FilterRuleConfigTableEntry 11 }	INTEGER forward(1) discard(2) none(3)	read-write	mandatory	N
sysDevL3FilterRuleDestPortNum	{ sysDevL3FilterRuleConfigTableEntry 12 }	INTEGER	read-write	mandatory	N
sydDevL3FilterRuleMaxDestPortNum	{ sysDevL3FilterRuleConfigTableEntry 13 }	INTEGER	read-write	mandatory	N
sysDevL3FilterRuleDestCompType	{ sysDevL3FilterRuleConfigTableEntry 14 }	INTEGER none(1) eq(2) neq(3) gt(4) lt(5) inRange(6) outRange(7)	read-write	mandatory	N
sysDevL3FilterRuleProtocolTypeUdp	{ sysDevL3FilterRuleConfigTableEntry 15 }	INTEGER forward(1) discard(2) none(3)	read-write	mandatory	N
sysDevL3FilterRuleProtocolTypeTcp	{ sysDevL3FilterRuleConfigTableEntry 16 }	INTEGER forward(1) discard(2) none(3)	read-write	mandatory	N
sysDevL3FilterRuleProtocolTypeIcmp	{ sysDevL3FilterRuleConfigTableEntry 17 }	INTEGER forward(1) discard(2) none(3)	read-write	mandatory	N
sysDevL3FilterRuleRowStatus	{ sysDevL3FilterRuleConfigTableEntry 18 }	RowStatus	read-write	mandatory	N

5.32.4.1.9.1 sysDevL3FilterRuleIndex

The value of this object is an index of the filter rule table. This object is used by sysDevFilterConfigTable (Section 5.32.4.1.3, Page 422) to identify the row in the table. When creating new row of this table the value of this object may be obtained by retrieving the value of sysDevL3FilterRuleIndexNext (Section 5.32.4.1.10, Page 431).

5.32.4.1.9.2 sysDevL3FilterRuleName

The value of this object corresponds directly to the name of the rule. It is highly recommended that the RuleName be unique user friendly name.

5.32.4.1.9.3 sysDevL3FilterRuleSrcAddress

The value of this object corresponds directly to a source IP Address. This object could be set if IP packets need to be filtered based on where it is coming from. Setting this object to 0.0.0.0 disables source IP address comparison.

5.32.4.1.9.4 sysDevL3FilterRuleSrcAddrMask

The value of this object corresponds directly to an source IP Address Mask. The value of this object is bitwise anded with the source address within the IP packet.

5.32.4.1.9.5 sysDevL3FilterRuleSrcAddrAction

The following values are defined for this object:

- forward(1)

- discard(2)
- none(3)

This object specifies action taken with the source address filtering. If the selection allows any SrcAddr (effectively ignoring this filed) than none(3) is valid choice for this object.

5.32.4.1.9.6 sysDevL3FilterRuleSrcPortNum

The UDP or TCP port number of the source on which this filter is applied. This object is used as a start of range if the source comparison type is in-range or out-range.

5.32.4.1.9.7 sysDevL3FilterRuleMaxSrcPortNum

The maximum UDP or TCP port number of the source on which this filter is applied. This object is valid only if the source comparison type is in-range or out-range.

5.32.4.1.9.8 sysDevL3FilterRuleSrcCompType

The value of this object corresponds to the type of comparison which is to be made for source based filtering for the particular rule of the filter.

The following values are defined for this object:

- none(1) - no comparison is to be made
- eq(2) - equal to
- neq(3) - not equal to
- gt(4) - greater than
- lt(5) - less than
- inRange(6) - the source port number of the TCP or UDP packet is within a specified range
- outRange(7) - the source port number of the TCP or UDP packet lies outside a specified range

5.32.4.1.9.9 sysDevL3FilterRuleDestAddress

The value of this object corresponds directly to an destination IP Address. This object could be set if IP packets need to be filtered based on where it is going to.

5.32.4.1.9.10 sysDevL3FilterRuleDestAddrMask

The value of this object corresponds directly to an destination IP Address Mask. The value of this object is bitwise anded with the destination address within the IP packet.

5.32.4.1.9.11 sysDevL3FilterRuleDestAddrAction

The following values are defined for this object:

- forward(1)
- discard(2)
- none(3)

This object specifies action taken for destination address filtering. If the selection allows any DestAddr (effectively ignoring this filed) than none(3) is valid choice for this object.

5.32.4.1.9.12 sysDevL3FilterRuleDestPortNum

The UDP or TCP port number of the destination on which this filter is applied. This object is used as a start of range if the source comparison type is in-range or out-range.

5.32.4.1.9.13 sysDevL3FilterRuleMaxDestPortNum

The maximum UDP or TCP port number of the destination on which this filter is applied. This object is valid only if the destination comparison type is in-range or out-range

5.32.4.1.9.14 sysDevL3FilterRuleDestCompType

The value of this object corresponds to the type of comparison which is to be made for destination based filtering for the particular rule of the filter.

The following values are defined for this object:

- none(1) - no comparison is to be made
- eq(2) - equal to
- neq(3) - not equal to
- gt(4) - greater than
- lt(5) - less than
- inRange(6) - the source port number of the TCP or UDP packet is within a specified range
- outRange(7) - the source port number of the TCP or UDP packet lies outside a specified range

5.32.4.1.9.15 sysDevL3FilterRuleProtocolTypeUdp

The following values are defined for this object:

- forward(1)
- discard(2)
- none(3)

The value of this object corresponds to the action taken on a particular UDP packet based on the source/ destination port comparison rule. If the selection allows any Udp packet (effectively ignoring this filed) than none(3) is valid choice for this object.

5.32.4.1.9.16 sysDevL3FilterRuleProtocolTypeTcp

The following values are defined for this object:

- forward(1)
- discard(2)
- none(3)

The value of this object corresponds to the action taken on a particular TCP packet bases on the source/ destination port comparison rule. If the selection allows any Tcp packet (effectively ignoring this filed) than none(3) is valid choice for this object.

5.32.4.1.9.17 sysDevL3FilterRuleProtocolTypeIcmp

The following values are defined for this object:

- forward(1)
- discard(2)
- none(3)

The value of this object corresponds to the action taken on a particular ICMP packet bases on the protocol type comparison rule. If the selection allows any Icmp packet (effectively ignoring this filed) than none(3) is valid choice for this object.

5.32.4.1.9.18 sysDevL3FilterRuleRowStatus

The object is used to add or delete a row from this table.

5.32.4.1.10 sysDevL3FilterRuleIndexNext

This object contains an appropriate value to be used for sysDevL3FilterRuleConfigTable (Section 5.32.4.1.9, Page 428). The value of 0 indicates that no unassigned entries are available.To obtain sysDevL3FilterRuleIndex value for the new entry the manager issues a management protocol retrieval operation to obtain a current value of this object. After each retrieval, the agent must modify the value to the next unassigned index to prevent assignment of the same

value to multiple management systems. A management systems should repeat the read to obtain a new value should an attempt to create the new row using previously returned index fail. As this is a scalar object if the mib walk is done on this object it would return the next unused value.

5.32.4.2 sysDevIpFilter

The IP Filter Group

OIDs:

- sysDevIpFilter - { sysDevFilterMIBObjects 2 } 1.3.6.1.4.1.1795.2.24.2.23.1.2

Table 304. sysDevIpFilter

Object	OID	Syntax	Access	Status	Supported
sysDevIpFilterConfigTable	{ sysDevIpFilter 1 }	SEQUENCE OF sysDevIpFilterConfigTableEntry	not-accessible	mandatory	N
sysDevIpFilterRuleConfigTable	{ sysDevIpFilter 2 }	SEQUENCE OF sysDevIpFilterRuleConfigTableEntry	not-accessible	deprecated	N
sysDevMaxNumOfInputIpFilters	{ sysDevIpFilter 3 }	INTEGER	read-only	mandatory	N
sysDevMaxNumOfOutoutIpFilters	{ sysDevIpFilter 4 }	INTEGER	read-only	mandatory	N
sysDevIpFilterBindingTable	{ sysDevIpFilter 5 }	SEQUENCE OF sysDevIpFilterBindingTableEntry	not-accessible	mandatory	N
sysDevIpFilterSNBindingTable	{ sysDevIpFilter 6 }	SEQUENCE OF sysDevIpFilterSNBindingTableEntry	not-accessible	mandatory	N
sysDevIpInputPacketsFiltered	{ sysDevIpFilter 7 }	Counter	read-only	mandatory	N
sysDevIpOutoputPacketsFiltered	{ sysDevIpFilter 8 }	Counter	read-only	mandatory	N

5.32.4.2.1 sysDevIpFilterConfigTable

This table allows the user to configure filtering information. Every filter is comprised of a number of rules (see SysDevRuleTable). When a filter is bound to an interface, all the rules of that particular filter apply to that particular interface. Also the same filter maybe bound to several interfaces. This table only allows us to configure information about a particular filter. this table is indexed by sysDevIpFilterName.

Index:

- sysDevIpFilterName

OIDs:

- sysDevIpFilterConfigTable - { sysDevIpFilter 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.2.1
- sysDevIpFilterConfigTableEntry - { sysDevIpFilterConfigTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.2.1.1

Table 305. sysDevIpFilterConfigTable

Object	OID	Syntax	Access	Status	Supported
sysDevIpFilterName	{ sysDevIpFilterConfigTableEntry 1 }	DisplayString(SIZE(1…12))	read-only	mandatory	N

Table 305. sysDevIpFilterConfigTable

Object	OID	Syntax	Access	Status	Supported
sysDevIpDefFilterAction	{ sysDevIpFilterConfigTableEntry 2 }	INTEGER forward(1) discard(2) delete(3)	read-write	mandatory	N
sysDevIpFilterNumOfDynamicRules	{ sysDevIpFilterConfigTableEntry 3 }	INTEGER	read-only	mandatory	N
sysDevIpFilterNumOfStaticRules	{ sysDevIpFilterConfigTableEntry 4 }	INTEGER	read-only	mandatory	N
sysDevIpFilterRefCount	{ sysDevIpFilterConfigTableEntry 5 }	INTEGER	read-only	mandatory	N
sysDevIpFilterTcpAckFilterAction	{ sysDevIpFilterConfigTableEntry 6 }	INTEGER forward(1) discard(2) noOp(3)	read-write	mandatory	N
sysDevIpFilterDhcpFilterAction	{ sysDevIpFilterConfigTableEntry 7 }	INTEGER forward(1) discard(2) noOp(3)	read-write	mandatory	N
sysDevIpFilterRowStatus	{ sysDevIpFilterConfigTableEntry 8 }	RowStatus	read-write	mandatory	N

5.32.4.2.1.1 sysDevIpFilterName

The value of this object corresponds directly to the name of the filter.

5.32.4.2.1.2 sysDevIpDefFilterAction

The value of this object corresponds to the default action taken on a particular IP packet based on the filter rule. The following values are defined for this object:

- forward(1) - the packet is forwarded
- discard(2) - the packet is discarded
- delete(3) - the rule of a particular filter or the filter as a whole is deleted

5.32.4.2.1.3 sysDevIpFilterNumOfDynamicRules

This object corresponds to the number of dynamic rules configured for this filter

5.32.4.2.1.4 sysDevIpFilterNumOfStaticRules

This object corresponds to the number of static rules configured for this filter

5.32.4.2.1.5 sysDevIpFilterRefCount

This object corresponds to the number of references for this filter by the various interfaces of the system For Example if filter “abc” is bound to two of the DSL interfaces as input filters. then the reference count for this filter is 2. if a filter “def” is bound to the 10BaseT interface as both an input and an output filter. The reference count for this filter would be 2. Filters can be deleted from the system (by a user) only if the reference count equals 0.

5.32.4.2.1.6 sysDevIpFilterTcpAckFilterAction

The value of this object corresponds to the default action taken on the first TCP ACK packet of the TCP connection. The following values are defined for this object:

- forward(1)
- discard(2)
- noOp(3)

This object is not supported for MCC.

5.32.4.2.1.7 sysDevIpFilterDhcpFilterAction

The value of this object corresponds to the default action taken on the DHCP transaction messages. The following values are defined for this object:

- forward(1)
- discard(2)
- noOp(3)

This object is not supported for mcc.

5.32.4.2.1.8 sysDevIpFilterRowStatus

The object is used to add or delete a row from this table.

5.32.4.2.2 sysDevIpFilterRuleConfigTable

This table has been deprecated.

5.32.4.2.3 sysDevMaxNumOfInputIpFilters

The value of this object corresponds directly to the max number of input IP filters supported by each device.

5.32.4.2.4 sysDevMaxNumOfOutputIpFilters

The value of this object corresponds directly to the max number of Output IP filters supported by each device.

5.32.4.2.5 sysDevIpFilterBindingTable

This table allows us to bind filters to a particular interface. As such this table is doubly indexed by ifIndex (Section 5.3.4.3.1, Page 37) and sysDevIpBindingFilterName. This table is a sequence of sysDevIpFilterBindingTableEntry where an entry describes a specific entry of the filter binding table.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- sysDevIpBindingFilterName (Section 5.32.4.2.5.1, Page 434)

OIDs:

- sysDevIpFilterBindingTable - { sysDevIpFilter 5 } 1.3.6.1.4.1.1795.2.24.2.23.1.2.5
- sysDevIpFilterBindingTableEntry - { sysDevIpFilterBindingTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.2.5.1

Table 306. sysDevIpFilterBindingTable

Object	OID	Syntax	Access	Status	Supported
sysDevIpBindingFilterName	{ sysDevIpFilterBindingTableEntry 1 }	DisplayString (SIZE(1…12))	read-only	mandatory	N
sysDevIpBindingFilterType	{ sysDevIpFilterBindingTableEntry 2 }	INTEGER inputFilter(1) outputFilter(2) inputOutputFilter(3)	read-write	mandatory	N
sysDevIpBindingFilterRowStatus	{ sysDevIpFilterBindingTableEntry 3 }	RowStatus	read-write	mandatory	N

5.32.4.2.5.1 sysDevIpBindingFilterName

The value of this object corresponds directly to the name of the filter.

5.32.4.2.5.2 sysDevIpBindingFilterType

The value of this object corresponds directly to the type of the filter which needs to be bound to a particular interface. The following values are defined for this object:

- inputFilter(1) - This indicates that the filter is applied to the packets coming in to the interface
- outputFilter(2) - this indicates that the filter is applied to packets going out the interface
- inputOutputFilter(3) - this indicates that the filter is applied packets going out the interface and to packets coming in to the interface

5.32.4.2.5.3 sysDevIpBindingFilterRowStatus

This object is used to add or delete a row from this table.

5.32.4.2.6 sysDevIpFilterSNBindingTable

This is the Service Node Filter bindings which allows us to bind filters to a particular interface on port cards. This table is indexed by ifIndex, Section 5.3.4.3.1, Page 37, sysDevIpSNBindingVnidId and sysDevIpSNBindingFilterName. This table is a sequence of sysDevIpFilterSNBindingTableEntry where an entry describes a specific entry of the SNfilter binding table.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- sysDevIpSNBindingVnidId
- sysDevIpSNBindingFilterName

OIDs:

- sysDevIpFilterSNBindingTable - { sysDevIpFilter 6 } 1.3.6.1.4.1.1795.2.24.2.23.1.2.6
- sysDevIpFilterSNBindingTableEntry - { sysDevIpFilterSNBindingTable 1 } 1.3.6.1.4.1.1795.2.24.2.23.1.2.6.1

Table 307. sysDevIpFilterSNBindingTableEntry

Object	OID	Syntax	Access	Status	Supported
sysDevIpSNBindingVnidId	{ sysDevIpFilterSNBindingTableEntry 1 }	VnidRange	read-only	mandatory	N
sysDevIpSNBindingFilterName	{ sysDevIpFilterSNBindingTableEntry 2 }	DisplayString (SIZE(1…12))	read-write	mandatory	N
sysDevIpSNBindingFilterType	{ sysDevIpFilterSNBindingTableEntry 3 }	INTEGER inputFilter(1) outputFilter(2) inputOutputFilter(3)	read-write	mandatory	N
sysDevIpSNBindingFilterRowStatus	{ sysDevIpFilterSNBindingTableEntry 4 }	RowStatus	read-write	mandatory	N

5.32.4.2.6.1 sysDevIpSNBindingVnidId

The VNID Id number of the virtual network to which this filter is bound.

5.32.4.2.6.2 sysDevIpSNBindingFilterName

The value of this object corresponds directly to the name of the filter.

5.32.4.2.6.3 sysDevIpSNBindingFilterType

The value of this object corresponds directly to the type of the filter which needs to be bound to a particular interface. The following values are defined for this object:

- inputFilter(1) - This indicates that the filter is applied to the packets coming in to the interface
- outputFilter(2) - this indicates that the filter is applied to packets going out the interface

- inputOutputFilter(3) - this indicates that the filter is applied packets going out the interface and to packets coming in to the interface

5.32.4.2.6.4 sysDevIpSNBindingFilterRowStatus

This object is used to add or delete a row from this table.

5.32.4.2.7 sysDevIpInputPacketsFiltered

The number of IP packets that have been filtered on all the incoming interfaces due to not matching programmed filter entries on interfaces that are activated for IP filtering.This is supported on MCC only.

5.32.4.2.8 sysDevIpOutputPacketsFiltered

The number of IP packets that have been filtered on all the outgoing interfaces due to not matching programmed filter entries on interfaces that are activated for IP filtering.This is supported on MCC only.

5.32.5 Traps

OIDs:

- sysDevFilterMIBTraps - { pdn-filter 2 } 1.3.6.1.4.1.1795.2.24.2.23.2

Table 308. PDN-FILTER-MIB Traps

Trap	ENTERPRISE	:=	Variable Bindings / Objects	Status	Supported
sysDevSNInjectionFailureTrap	sysDevFilterMIBTraps	22	ifIndex sysDevSNInjectionVnid sysDevSNInjectionType	current	N
sysDevSNInjectionIncompatibleTrap	sysDevFilterMIBTraps	23	ifIndex sysDevSNInjectionVnid sysDevSNInjectionType	current	N

5.32.5.1 sysDevSNInjectionFailureTrap

This trap signifies that the injection of filters to SN failed It is of warning class.

5.32.5.2 sysDevSNInjectionIncompatibleTrap

This trap signifies that the injection of filters is not supported by the SN. It is of warning class.

5.33 PDN-IF-EXT-CONFIG-MIB — pdn_IfExtConfig.mib

5.33.1 Introduction

This Mib is created to facilitate configuration of interface related objects.

The version of the MIB used for this document had a LAST-UPDATED of 0005030000Z which decodes to May 3, 2000.

5.33.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-interface - { pdn-common 6 } 1.3.6.1.4.1.1795.2.24.2.6
- pdnIfExt - { pdn-interface 12 } 1.3.6.1.4.1.1795.2.24.2.6.12
- pdnIfExtEncapConfig - { pdnIfExt 3 } 1.3.6.1.4.1.1795.2.24.2.6.12.3

Table 309. PDN-IF-EXT-CONFIG-MIB Objects

Object	OID	Description	Supported
pdnIfExtEncapConfig	{ pdnIfExt 3 }	Interface Configuration Group	Y

5.33.2.1 pdnIfExtEncapConfig

The interface configuration group.

OIDs:

- pdnIfExtEncapConfig - { pdnIfExt 3 } 1.3.6.1.4.1.1795.2.24.2.6.12.3

Table 310. pdnIfExtEncapConfig

Object	OID	Syntax	Access	Status	Supported
pdnIfMultiprotocolEncapConfigTable	{ pdnIfExtEncapConfig 1 }	SEQUENCE OF pdnIfMultiprotocolEncapConfigEntry	not-accessible	mandatory	Y

5.33.2.1.1 pdnIfMultiprotocolEncapConfigTable

This table that contains additional interface configuration information.There is one of these rows for each interface that supports RFC 1483.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

pdnIfMultiprotocolEncapConfigTablepdnIfExtEncapConfig1.3.6.1.4.1.1795.2.24.2.6.12.3.1pdnIfMultiprotocolEncapConfigEntry-pdnIfMultiprotocolEncapConfigTable1.3.6.1.4.1.1795.2.24.2.6.12.3.1.1

Table 311. pdnIfMultiprotocolEncapConfigTable

Object	OID	Syntax	Access	Status	Supported
pdnIfMultiprotocolEncapConfigIPRoutedPDUs	{ pdnIfMultiprotocolEncapConfigEntry 1 }	INTEGER none(1) llcSnap(2) vcBasedMultiplexing(3)	read-write	current	Y
pdnIfMultiprotocolEncapConfigBridgedPDUs	{ pdnIfMultiprotocolEncapConfigEntry 2 }	INTEGER none(1) llcSnap(2) vcBasedMultiplexing(3)	read-write	current	Y

5.33.2.1.1.1 pdnIfMultiprotocolEncapConfigIPRoutedPDUs

The following values are defined for this object:

- none(1)
- llcSnap(2)
- vcBasedMultiplexing(3)

In the upstream direction, the user can configure the IP routed PDUs in the LLC SNAP encapsulation or VC Based Multiplexing encapsulation (RFC 1483). If neither is configured “none” is used. The direction is determined by the transmit direction. Initially this object was developed for the endpoint use where the transmit direction is upstream. This object could be used for CO type equipment where the direction would be downstream.

5.33.2.1.1.2 pdnIfMultiprotocolEncapConfigBridgedPDUs

The following values are defined for this object:

- none(1)
- llcSnap(2)
- vcBasedMultiplexing(3)

In the transmit direction, the user can configure the IP routed PDUs in the LLC SNAP encapsulation or VC Based Multiplexing encapsulation (RFC 1483). If neither is configured “none” is used.

5.34 PDN-IGMP-STD-EXT-MIB — PDN-IGMP-STD-EXT-MIB.mib

5.34.1 Introduction

This MIB module is a supplement to the IGMP-STD-MIB, RFC 2933.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200409200000Z,” or September 10, 2004.

5.34.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.34.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnIgmpStdExtMIB - { pdn-common 47 } 1.3.6.1.4.1.1795.2.24.2.47
- pdnIgmpStdExtNotifications - { pdnIgmpStdExtMIB 0 } 1.3.6.1.4.1.1795.2.24.2.47.0
- pdnIgmpStdExtObjects - { pdnIgmpStdExtMIB 1 } 1.3.6.1.4.1.1795.2.24.2.47.1
- pdnIgmpStdExtAFNs - { pdnIgmpStdExtMIB 2 } 1.3.6.1.4.1.1795.2.24.2.47.2

5.34.4 MIB Objects

Table 312. PDN-IGMP-STD-EXT-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnIgmpInterfaceExtTable	{ pdnIgmpStdExtObjects 1 }	SEQUENCE OF pdnIgmpInterfaceExtEntry	not-accessible	current	Y
pdnIgmpCacheExtTable	{ pdnIgmpStdExtObjects 2 }	SEQUENCE OF pdnIgmpCacheExtEntry	not-accessible	current	Y
pdnIgmpSnoopingSelection	{ pdnIgmpStdExtObjects 3 }	SwitchState	read-write	current	Y
pdnIgmpGeneralQuery-Interval	{ pdnIgmpStdExtObjects 4 }	Unsigned32	read-write	current	Y
pdnIgmpQuerierAutoDetect	{ pdnIgmpStdExtObjects 5 }	SwitchState	read-write	current	Y
pdnIgmpQuerierDot1d-BridgePort	{ pdnIgmpStdExtObjects 6 }	Unsigned32(1…65535)	read-write	current	Y

5.34.4.1 pdnIgmpInterfaceExtTable

This table extends the igmpInterfaceTable.

Augments:

- igmpInterfaceTable (Section 5.18.4.1, Page 303)

Index:

- igmpInterfaceIfIndex (Section 5.6.5.1.1.1, Page 87)

OIDs:

- pdnIgmpInterfaceExtTable - { pdnIgmpStdExtObjects 1 } 1.3.6.1.4.1.1795.2.24.2.47.1.1
- pdnIgmpInterfaceExtEntry - { pdnIgmpInterfaceExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.47.1.1.1

Table 313. pdnIgmpInterfaceExtTable

Object	OID	Syntax	Access	Status	Supported
pdnIgmpInterface SnoopEnableDisable	{ pdnIgmpInterfaceExtEntry 1 }	SwitchState	read-write	current	Y
pdnIgmpInterface LeaveDelay	{ pdnIgmpInterfaceExtEntry 2 }	Unsigned32(0…255)	read-write	current	Y
pdnIgmpInterface LeaveJoinForwardingDelay	{ pdnIgmpInterfaceExtEntry 3 }	Unsigned32(0…255)	read-write	current	Y

5.34.4.1.1 pdnIgmpInterfaceSnoopEnableDisable

This object specifies whether or not IGMP Snooping is enabled or disabled. The following are valid values:

- enabled(1) - IGMP Snooping is Enabled
- disabled(2) - IGMP Snooping is Disabled

DEFVAL-disabled(2)

5.34.4.1.2 pdnIgmpInterfaceLeaveDelay

This object specifies the leave-delay period in units of tenths of a second.

This value is defined as the period of time during which a multicast stream will continue to be forwarded on an interface after receipt of an IGMP leave message for that stream.

DEFVAL-3

5.34.4.1.3 pdnIgmpInterfaceLeaveJoinForwardingDelay

This object specifies the leave-join-delay period in units of tenths of a second.

This value is defined as the period of time during which new multicast streams will not be forwarded on a port following the receipt of an IGMP Leave message.

DEFVAL -3

5.34.4.2 pdnIgmpCacheExtTable

This table extends the igmpCacheTable.

Augments:

- igmpCacheTable (Section 5.18.4.2, Page 306)

Indicies:

- igmpCacheAddress (Section 5.18.4.2.1, Page 306)
- igmpCacheIfIndex (Section 5.18.4.2.2, Page 306)

OIDs:

- pdnIgmpCacheExtTable - { pdnIgmpStdExtObjects 2 } 1.3.6.1.4.1.1795.2.24.2.47.1.2
- pdnIgmpCacheExtEntry - { pdnIgmpCacheExtTable 1 } 1.3.6.1.4.1.1795.2.24.2.47.1.2.1

Table 314. pdnIgmpCacheExtTable

Object	OID	Syntax	Access	Status	Supported
pdnIgmpCacheStatsMulticast-PktsIn	{ pdnIgmpCacheExtEntry 1 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsMulticast-PktsOut	{ pdnIgmpCacheExtEntry 2 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsIgmpQueries-In	{ pdnIgmpCacheExtEntry 3 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsIgmpQueries-Out	{ pdnIgmpCacheExtEntry 4 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsIgmpReports-In	{ pdnIgmpCacheExtEntry 5 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsIgmpReports-Out	{ pdnIgmpCacheExtEntry 6 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsIgmpLeaves-In	{ pdnIgmpCacheExtEntry 7 }	Counter32	read-only	current	Y
pdnIgmpCacheStatsIgmpLeaves-Out	{ pdnIgmpCacheExtEntry 8 }	Counter32	read-only	current	Y
pdnIgmpCacheLastReporterMAC	{ pdnIgmpCacheExtEntry 9 }	Counter32	read-only	current	Y

5.34.4.2.1 pdnIgmpCacheStatsMulticastPktsIn

The total number of IP Multicast packets received excluding IGMP messages.

5.34.4.2.2 pdnIgmpCacheStatsMulticastPktsOut

The total number of IP Multicast packets transmitted excluding IGMP messages.

5.34.4.2.3 pdnIgmpCacheStatsIgmpQueriesIn

The total number of IGMP Query messages received.

5.34.4.2.4 pdnIgmpCacheStatsIgmpQueriesOut

The total number of IGMP Query messages transmitted.

5.34.4.2.5 pdnIgmpCacheStatsIgmpReportsIn

The total number of IGMP Report messages received.

5.34.4.2.6 pdnIgmpCacheStatsIgmpReportsOut

The total number of IGMP Report messages transmitted.

5.34.4.2.7 pdnIgmpCacheStatsIgmpLeavesIn

The total number of IGMP Leave messages received.

5.34.4.2.8 pdnIgmpCacheStatsIgmpLeavesOut

The total number of IGMP Leave messages transmitted.

5.34.4.2.9 pdnIgmpCacheLastReporterMAC

This object displays the MAC address of the source of the last membership report received for this IP Multicast group address on this interface. If no membership report has been received, this object has the value 0.

5.34.4.2.10 pdnIgmpSnoopingSelection

This object specifies whether or not IGMP Snooping is enabled or disabled on a system-wide basis.

- enabled(1) - IGMP Snooping is Enabled on the device
- disabled(2) - IGMP Snooping is Disabled on the device

The MIB recommended default is disabled(2).

5.34.4.2.11 pdnIgmpGeneralQueryInterval

This object is similar to the igmpInterfaceQueryInterval object defined in RFC2933. However, it can be used to set a global value that applies to all the interfaces in the system with IGMP enabled. The units of the object are “seconds.”

A value of zero disables this option and thus the agent shall employ the value configured in the igmpInterfaceQueryInterval for each interface running IGMP. In the other hands, if this object is set to a value greater than zero, the value takes precedence over the value configured in the igmpInterfaceQueryInterval for that interface.

The MIB recommended default is 125 seconds.

5.34.4.2.12 pdnIgmpQuerierAutoDetect

This object is used to auto-detect the bridge port from which the IGMP Querier can be reached by the device. Thus, only 1 IGMP Querier can be configured.

When this object is set to “disabled,” the value currently configured in pdnIgmpQuerierDot1BridgePort is used to access the IGMP Querier.

When this object is set to “enabled,” the agent automatically detects the bridge port in the system the permits access to the IGMP Querier. The discovered bridge port will overwrites the value previously set in the pdnIgmpQuerierDot1BridgePort.

5.34.4.2.13 pdnIgmpQuerierDot1dBridgePort

This object is the bridge port number from which this IGMP Querier can be accessed.

The value of this object will be overwritten by the agent if the pdnIgmpQuerierAutoDetect is enabled.

NMS/EMS should make use of existing mibs that provide additional information on the bridge port assignments/ mapping in the device.

5.34.5 Conformance Information

5.34.5.1 Compliance

5.34.5.1.1 pdnIgmpStdExtMIBCompliance

The **deprecated** compliance statement for IP Multicast products which implement the pdnIpMcaseMIB.

Please refer to pdnIgmpStdExtMIBComplianceV2 below.

Mandatory Groups:

None specified.

Optional Groups:

- pdnIgmpStdExtConfigGroup - This group is mandatory only for implementations which the relative parameters to be configured for IP Multicast and IGMP operation.
- pdnIgmpStdExtStatsGroup - This group is mandatory only for implementations which require additional IP Multicast statistics.

Objects that have a MIN-ACCESS defined:

None specified

5.34.5.1.2 pdnIgmpStdExtMIBComplianceV2

The compliance statement for IP products which implement the IGMP protocol.

Mandatory Groups:

None specified.

Optional Groups:

- pdnIgmpStdExtConfigGroup - This group is mandatory only for implementations which the relative parameters to be configured for IP Multicast and IGMP operation.
- pdnIgmpStdExtStatsGroupV2 - This group is mandatory only for implementations which require additional IP Multicast statistics.
- pdnIgmpStdExtGeneralConfigGroupV2 - This group is mandatory only for implementations which the relative parameters to be configured for IP Multicast and IGMP operation.

5.34.5.2 Conformance

Table 315. PDN-IGMP-STD-EXT-MIB Conformance

Group	Description	Objects	Supported
pdnIgmpStdExtConfigGroup	Objects grouped for configuring IP Multicast and IGMP parameters.	pdnIgmpInterfaceSnoopEnableDisable	Y
		pdnIgmpInterfaceLeaveDelay	Y
		pdnIgmpInterfaceLeaveJoinForwardingDelay	Y
pdnIgmpStdExtStatsGroup (deprecated in favor of pdnIgmpStdExtStatsGroupV2)	Objects grouped for IP Multicast and IGMP Statistics.	pdnIgmpCacheStatsMulticastPktsIn	Y
		pdnIgmpCacheStatsMulticastPktsOut	Y
		pdnIgmpCacheStatsIgmpQueriesIn	Y
		pdnIgmpCacheStatsIgmpQueriesOut	Y
		pdnIgmpCacheStatsIgmpReportsIn	Y
		pdnIgmpCacheStatsIgmpReportsOut	Y
		pdnIgmpCacheStatsIgmpLeavesIn	Y
		pdnIgmpCacheStatsIgmpLeavesOut	Y
pdnIgmpStdExtGeneralConfigGroup (deprecated in favor of pdnIgmpStdExtGeneralConfig-GroupV2)	Objects grouped for general configuring of IP Multicast and IGMP parameters.	pdnIgmpSnoopingSelection	Y
pdnIgmpStdExtStatsGroupV2	Objects grouped for IP Multicast and IGMP Statistics.	pdnIgmpCacheStatsMulticastPktsIn	Y
		pdnIgmpCacheStatsMulticastPktsOut	Y
		pdnIgmpCacheStatsIgmpQueriesIn	Y
		pdnIgmpCacheStatsIgmpQueriesOut	Y
		pdnIgmpCacheStatsIgmpReportsIn	Y
		pdnIgmpCacheStatsIgmpReportsOut	Y
		pdnIgmpCacheStatsIgmpLeavesIn	Y
		pdnIgmpCacheStatsIgmpLeavesOut	Y
		pdnIgmpCacheLastReporterMAC	Y
pdnIgmpStdExtGeneralConfigGroupV2	Objects grouped for general configuring of IP Multicast and IGMP parameters.	pdnIgmpSnoopingSelection	Y
		pdnIgmpGeneralQueryInterval	Y
		pdnIgmpQuerierAutoDetect	Y
		pdnIgmpQuerierDot1dBridgePort	Y

5.35 PDN-INET-MIB — pdn_inet.mib

5.35.1 Introduction

This MIB is used to configure IP address of some interfaces.

The version of the MIB used for this document had a LAST-UPDATED of 0202210000Z which decodes to February 21, 2002.

5.35.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.35.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-inet - { pdn-common 26 } 1.3.6.1.4.1.1795.2.24.2.26
- pdnInetMIBObjects - { pdn-inet 1 } 1.3.6.1.4.1.1795.2.24.2.26.1
- pdnInetMIBTraps - { pdn-inet 2 } 1.3.6.1.4.1.1795.2.24.2.26.2

5.35.4 MIB Objects

Table 316. PDN-INET-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnInetTelnetServerPort	{ pdnInetMIBObjects 1 }	INTEGER(1…65535)	read-write	current	N
pdnInetFtpServerControlPort	{ pdnInetMIBObjects 2 }	INTEGER(1…65535)	read-write	current	N
pdnInetFtpServerDataPort	{ pdnInetMIBObjects 3 }	INTEGER(1…65535)	read-write	current	N
pdnInetIpAddressTableMaxIpSubnets	{ pdnInetMIBObjects 4 }	Integer32	read-only	current	N
pdnInetIpAddressTableCurrentIpSubnets	{ pdnInetMIBObjects 5 }	Integer32	read-only	current	N
pdnInetIpAddressTable	{ pdnInetMIBObjects 6 }	SEQUENCE OF pdnInetIpAddressTableEntry	not-accessible	current	Y

5.35.4.1 pdnInetTelnetServerPort

This object contains the telnet server (daemon) tcp port number.

The default value for this object is 23.

5.35.4.2 pdnInetFtpServerControlPort

This object contains the ftp server control (daemon) tcp port number.

The default value for this object is 21.

5.35.4.3 pdnInetFtpServerDataPort

This object contains the ftp server data connection tcp port number.

The default value for this object is 20.

5.35.4.4 pdnInetIpAddressTableMaxIpSubnets

This object indicates the maximum number of entries in the pdnInetAddressTable.

5.35.4.5 pdnInetIpAddressTableCurrentIpSubnets

This object indicates the number of entries configured in the pdnInetAddressTable.

5.35.4.6 pdnInetIpAddressTable

This table used to configure IP addresses for an interface which support the internet (IP) protocol. This table is a sequence of pdnInetIpAddressTableEntry where each entry corresponds to an entry in the IP address table.

Indices:

- ifIndex (Section 5.3.4.3.1, Page 37)
- pdnInetIpAddress

OIDs:

- pdnInetIpAddressTable - { pdnInetMIBObjects 6 } 1.3.6.1.4.1.1795.2.24.2.26.1.6
- pdnInetIpAddressTableEntry - { pdnInetIpAddressTable 1 } 1.3.6.1.4.1.1795.2.24.2.26.1.6.1

Table 317. pdnInetIpAddressTable

Object	OID	Syntax	Access	Status	Supported
pdnInetIpAddress	{ pdnInetIpAddressTableEntry 1 }	IpAddress	not-accessible	current	Y
pdnInetIpSubnetMask	{ pdnInetIpAddressTableEntry 2 }	IpAddress	read-create	current	Y
pdnInetIpAddressType	{ pdnInetIpAddressTableEntry 3 }	INTEGER primary(1) secondary(2) primaryBootp(3) secondaryBootp(4)	read-create	current	N
pdnInetIpRowStatus	{ pdnInetIpAddressTableEntry 4 }	RowStatus	read-create	current	Y
pdnInetIpGateway	{ pdnInetIpAddressTableEntry 5 }	IpAddress	read-create	current	N

5.35.4.6.1 pdnInetIpAddress

This object is used to configure an IP address for an interface.

5.35.4.6.2 pdnInetIpSubnetMask

This object is used to configure an IP subnet mask for an interface. Subnet mask should result in contiguous subnets.

5.35.4.6.3 pdnInetIpAddressType

This object is used to configure the address type for an interface an interface can have only one primary ip address and more than one secondary ip address. The primary ip address is essentially the router ID. The secondary ip address would result in the interface being multi-homed. The following values are defined for this object:

- primary(1)
- secondary(2)

- primaryBootp(3)
- secondaryBootp(4)

The default value is primary(1).

5.35.4.6.4 pdnInetIpRowStatus

This object is used to manage rows (add/delete/modify) in this table.

5.35.4.6.5 pdnInetIpGateway

This object is used to configure an gateway for an interface.

5.36 PDN-IP-MULTICAST-MIB — PDN-IP-MULTICAST-MIB.mib

5.36.1 Introduction

This MIB contains the objects pertaining to the configuration and maintenance of the Internet Group Management Protocol (IGMP) and IP Multicast related operation in Paradyne devices.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200304240000Z,” or April 24, 2003.

5.36.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.36.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnIpMcastMIB - { pdn-common 45 } 1.3.6.1.4.1.1795.2.24.2.48
- pdnIpMcastNotifications - { pdnIpMcastMIB 0 } 1.3.6.1.4.1.1795.2.24.2.48.0
- pdnIpMcastObjects - { pdnIpMcastMIB 1 } 1.3.6.1.4.1.1795.2.24.2.48.1
- pdnIpMcastAFNs - { pdnIpMcastMIB 2 } 1.3.6.1.4.1.1795.2.24.2.48.2

- pdnIgmpProxy - { pdnIpMcastMIB 1 } 1.3.6.1.4.1.1795.2.24.2.48.1.1

5.36.4 MIB Objects

Table 318. PDN-IP-MULTICAST-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnIgmpProxy EnableDisable	{ pdnIgmpProxy 1 }	SwitchState	read-write	current	Y
pdnIgmpProxy ReportSummaryEnableDisable	{ pdnIgmpProxy 2 }	SwitchState	read-write	current	Y

5.36.4.1 pdnIgmpProxyEnableDisable

This object enables or disables IGMP Proxy. The following are valid values:

- enabled(1)
- disabled(2)

DEFVAL- disabled(2)

5.36.4.2 pdnIgmpProxyReportSummaryEnableDisable

This object enables or disables the summarization of all report messages into a single report message.The following are valid values:

- enabled(1)
- disabled(2)

DEFVAL- disabled(2)

5.36.5 Conformance Information

5.36.5.1 Compliance

5.36.5.1.1 pdnIpMcastMIBCompliance

The compliance statement for IP Multicast products which implement the pdnIpMcaseMIB.

Mandatory Groups:

None specified.

Optional Groups:

- pdnIgmpProxyGroup - This group is mandatory only for implementations which require IGMP Proxy.

Objects that have a MIN-ACCESS defined:

None specified

5.36.5.2 Conformance

Table 319. PDN-ENT-SENSOR-EXT-MIB Conformance

Group	Description	Objects	Supported
pdnIgmpProxyGroup	Objects grouped for IGMP Proxy.	pdnIgmpProxyEnableDisable	Y
		pdnIgmpProxyReportSummaryEnableDisable	Y

5.37 PDN-PROTO-SPEC-FILTER-MIB — PDN-PROTO-SPEC-FILTER-MIB.mib

5.37.1 Introduction

This MIB module contains objects pertaining to setting up protocol specific filters.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200312160000Z,” or December 16, 2003.

5.37.2 Textual Conventions

The following textual conventions are made in the reference document:

- PdnProtoSpecFilterList - A set or list of protocols to be filtered with one bit per protocol.
  - SYNTAX BITS
    - netbios(0) : Bit 00 Microsoft NetBIOS

5.37.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnProtoSpecFilterMIB - { pdn-common 49 } 1.3.6.1.4.1.1795.2.24.2.49
- pdnProtoSpecFilterNotifications - { pdnVlanMIB 0 } 1.3.6.1.4.1.1795.2.24.2.49.0
- pdnProtoSpecFilterObjects - { pdnVlanMIB 1 } 1.3.6.1.4.1.1795.2.24.2.49.1

5.37.4 MIB Objects

Table 320. PDN-PROTO-SPEC-FILTER-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnProtoSpecFilterPerDeviceList	{ pdnProtoSpecFilterObjects 1 }	PdnProtoSpecFilterList	read-write	current	Y
pdnProtoSpecFilterPerPortTable	{ pdnProtoSpecFilterObjects 2 }	SEQUENCE OF pdnProtoSpecFilterEntry	not-accessible	current	N

5.37.4.1 pdnProtoSpecFilterPerDeviceList

This object contains a list of protocols to be filtered on a per device/unit/chassis basis. When the bit is set for the respective protocol, it signifies to deny traffic of that protocol type (frames/packets are discarded). When the bit is reset for the respective protocol, it signifies to permit traffic of that protocol type (frames/packets are forwarded).

The pdnProtoSpecFilterPerPortList below is a list of protocols to be filtered on a per-port basis. Protocols listed in this object take precedence over protocols listed in the per-port object below if both objects are implemented.

5.37.4.2 pdnProtoSpecFilterPerPortTable

This object contains a list of protocols to be filtered on a per port basis. Each entry applies a filter to a particular port or interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- pdnProtoSpecFilterPerPortTable - { pdnProtoSpecFilterObjects 2 } 1.3.6.1.4.1.1795.2.24.2.49.1.2
- pdnProtoSpecFilterPerPortEntry - { pdnProtoSpecFilterPerPortTable 1 } 1.3.6.1.4.1.1795.2.24.2.49.1.2.1

Table 321. pdnProtoSpecFilterPerPortTable

Object	OID	Syntax	Access	Status	Supported
pdnProtoSpecFilterPerPortList	{ pdnProtoSpecFilterPerPortEntry 1 }	PdnProtoSpecFilterList	read-write	current	N

5.37.4.2.1 pdnProtoSpecFilterPerPortList

This object contains a list of protocols to be filtered on a per-port basis. When the bit is set for the respective protocol, it signifies to deny traffic of that protocol type (frames/packets are discarded). When the bit is reset for the respective protocol, it signifies to permit traffic of that protocol type (frames/packets are forwarded).

The pdnProtoSpecFilterPerDeviceList object above filters protocols on a “global” basis. That is per device/unit/chassis. The listed in the per-device object above take precedence over protocols listed in the per-port object here if both objects are implemented.

5.37.5 Compliance Statements

5.37.5.1 Compliance

5.37.5.1.1 pdnProtoSpecFilterMIBCompliance

The compliance statement for the pdnProtoSpecFilter entities which implement the pdnProtoSpecFilterMIB.

Mandatory Groups

None specified.

Optional Groups

- pdnProtoSpecFilterPerDeviceGroup - This group is mandatory only in those entities which implement filtering on a per-device basis.
- pdnProtoSpecFilterPerPortGroup - This group is mandatory only in those entities which implement filtering on a per-port basis.

5.37.5.2 Conformance

Table 322. PDN-PROTO-SPEC-FILTER-MIB Conformance

Group	Description	Objects	Supported
pdnProtoSpecFilter-PerDeviceGroup	Objects grouped for applying protocol-specific filters on a per-device basis.	pdnProtoSpecFilterPerDeviceList	Y
pdnProtoSpecFilter-PerPortGroup	Objects grouped for applying protocol-specific filters on a per-port basis.	pdnProtoSpecFilterPerPortList	N

5.38 PDN-REACHDSL-MIB — pdn_reachDSL.mib

5.38.1 Introduction

The Paradyne enterprise ReachDSL MIB. This MIB is written specifically to facilitate in the management of ReachDSL devices as well as help in configuration of Spectrum Management specific objects for ReachDSL products.

The version of the MIB used for this document had a LAST-UPDATED of 200210151700Z which decodes to October 15, 2002.

5.38.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.38.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2
- pdn-interfaces - { pdn-common 6 } 1.3.6.1.4.1.1795.2.24.2.6

- pdnReachDSL - { pdn-interfaces 20 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.20
- pdnReachDSLObjects - { pdnReachDSL 1 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1

5.38.4 MIB Objects

Table 323. PDN-REACHDSL-MIB Objects (1 of 2)

Object	OID	Syntax	Access	Status	Supported
reachDSLSpectrumMgmt Selection	{ pdnReachDSLObjects 1 }	INTEGER enabled(1) disabled(2)	read-write	deprecated[a]	N
reachDSLSpectrumMgmt Zone	{ pdnReachDSLObjects 2 }	INTEGER usa1(1) uk1(2) canada1(3) japan1(4) emea1(5)	read-only	deprecated	N
reachDSLSpectrumMgmt ConfTable	{ pdnReachDSLObjects 3 }	SEQUENCE of reachDSLSpectrumMgmtConfEntry	not-accessible	current	Y
reachDSLSpectrumMgmt LineInfoTable	{ pdnReachDSLObjects 4 }	SEQUENCE of reachDSLSpectrumMgmtLineInfoTable	not-accessible	deprecated[a.]	N

Table 323. PDN-REACHDSL-MIB Objects (2 of 2)

Object	OID	Syntax	Access	Status	Supported
reachDSLLineTable	{ pdnReachDSLObjects 5 }	SEQUENCE of reachDSLLineEntry	not-accessible	current	Y
reachDSLSpectrumMgmt LoopMeasurementMethod	{ pdnReachDSLObjects 6 }	INTEGER none(1) loopLength(2) ewl(3) quadMode(4)	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt EWLUnits	{ pdnReachDSLObjects 7 }	INTEGER none(1) feet(2) meters(3)	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt Mode	{ pdnReachDSLObjects 8 }	INTEGER enableOnly(1) disableOnly(2) both(3)	read-only	deprecated[a.]	N

a. The object's status states "current," but there is a note in the description that states that the object is deprecated.

5.38.4.1 reachDSLSpectrumMgmtSelection

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The following are valid values for this object:

- enabled(1)
- disabled(2)

This object is used to Enable or Disable Spectrum Management on this device. Enabling this object limits the DSL speeds and the maximum transmit power on each port based on line length in order to meet local Spectrum Management guidelines.

The values that this object can accept are controlled by the "Mode" object.

5.38.4.2 reachDSLSpectrumMgmtZone

The following are valid values for this object:

- usa1(1)
- uk1(2)
- canada1(3)
- japan1(4)
- emea1(5)

This object shows the geographic location or the spectrum zone used for spectrum management. It also represents which Spectrum Management standard is used on this device to calculate the effective maximum "Tx Rates" and the maximum allowed "Tx Power" to comply with Spectrum Management regulations for that specific region.

Note: Spectrum Management must be Enabled for "SpectrumMgmtZone" to take effect.

The MIB recommended default values is usal(1).

5.38.4.3 reachDSLSpectrumMgmtConfTable

The ReachDSL Spectrum Management Configuration Table This table is used for configuring spectrum management objects.

Note: This entry can be accessed only for ifIndex values that belong to a physical interface, for example, for ifType = “adsl.”

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- reachDSLSpectrumMgmtConfTable -{ pdnReachDSLObjects 3 }
  1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1.3
- reachDSLSpectrumMgmtConfEntry - { reachDSLSpectrumMgmtConfTable 1 }
  1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1.3.1

Table 324. reachDSLSpectrumMgmtConfTable

Object	OID	Syntax	Access	Status	Supported
reachDSLSpectrumMgmtConfEWL	{ reachDSLSpectrumMgmtConfEntry 1 }	Unsigned32	read-write	deprecated[a]	N
reachDSLSpectrumMgmtConfLoopLength	{ reachDSLSpectrumMgmtConfEntry 2 }	INTEGER short(1) medium(2) long(3)	read-write	deprecated[a.]	N
reachDSLSpectrumMgmtConfAtucMaxTxPower	{ reachDSLSpectrumMgmtConfEntry 3 }	INTEGER (-140…120)	read-write	current	Y
reachDSLSpectrumMgmtConfAturMaxTxPower	{ reachDSLSpectrumMgmtConfEntry 4 }	INTEGER (-140…120)	read-write	current	Y
reachDSLSpectrumMgmtConfQuadMode	{ reachDSLSpectrumMgmtConfEntry 5 }	INTEGER sameQuad(1) segregatedQuadUpto3km(2) segregatedQuadAbove3km(3)	read-write	deprecated[a.]	N

a. The object’s status states “current,” but there is a note in the description that states that the object is deprecated.

5.38.4.3.1 reachDSLSpectrumMgmtConfEWL

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The Equivalent Working Length (EWL). The value selected corresponds to the estimated length of the line in:

- feet - if “EWLUnits” object is “feet”
- meters - if “EWLUnits” object is “meters”

This value is usually specified by the carrier and MUST be in integral multiples of:

- 500 feet, if “EWLUnits” is “feet”
- 200 meters, if 'EWLUnits' is “meters”

The range of values that this object can take is determined by the “MinEWL” and “MaxEWL” objects. Changing this object may limit the maximum transmit power and the effective DSL speeds at which the ports train up.

Note: This object is used only if Spectrum Management is enabled and “LoopMeasurementMethod” is “ewl.”

5.38.4.3.2 reachDSLSpectrumMgmtConfLoopLength

This object shares a mutual exclusion relationship with “ConfLoopLength” & “ConfQuadMode” objects, as only one of the three is used for configuration at any given instant of time.”

The following are valid values for this object:

- short(1)
- medium(2)
- long(3)

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This entry specifies the loop length for the line. The configurable values are short(1), medium(2) and long(3). The “Loop Length” feature can be used only if the “reachDSLSpectrumMgmtSelection” object is Enabled and the “LoopMeasurementMethod” is “loopLength.” Changing this object may limit the maximum transmit power and the effective DSL speeds at which ports train up.

Note: This object shares a mutual exclusion relationship with the “ConfEWL” & “ConfQuadMode” objects: that is, only one of the three can be used for configuration at one time.

5.38.4.3.3 reachDSLSpectrumMgmtConfAtucMaxTxPower

The maximum transmit power at the ATUC (in tenths of a decibel). This value is based on the spectrum management selection. It ranges from -140 to 120 in increments of 10.

5.38.4.3.4 reachDSLSpectrumMgmtConfAturMaxTxPower

The maximum transmit power at the ATUR (in tenths of a decibel). This value is based on the spectrum management selection. It ranges from -140 to 120 in increments of 10.

5.38.4.3.5 reachDSLSpectrumMgmtConfQuadMode

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The following are valid values for this object:

- sameQuad(1)
- segregatedQuadUpto3km(2)
- segregatedQuadAbove3km(3)

This entry specifies the quad cable configuration & length for the line. The configurable values are sameQuad(1), segregatedQuadUpto3km(2) and segregatedQuadAbove3km(3). The “Quad Mode” feature can be used only if the “reachDSLSpectrumMgmtSelection” object is Enabled and the “LoopMeasurementMethod” is “quadMode.” Changing this object may limit the maximum transmit power and the effective DSL speeds at which ports train up.

Note: This object shares a mutual exclusion relationship with the “ConfEWL” & “ConfLoopLength” objects: only one of the three can be used for configuration at one time.

5.38.4.4 reachDSLSpectrumMgmtLineInfoTable

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The ReachDSL Spectrum Management Line Info Table. This table gives information about the spectrum management ranges that are applicable for each port, based on the spectral requirements for the device.

Note: This entry can be accessed only for ifIndex values that belong to a physical interface, for example, for ifType = “adsl.”

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- reachDSLSpectrumMgmtLineInfoTable - { pdnReachDSLObjects 4 }
  1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1.4
- reachDSLSpectrumMgmtLineInfoEntry - { reachDSLSpectrumMgmtLineInfoTable 1 }
  1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1.4.1

Table 325. reachDSLSpectrumMgmtLineInfoTable

Object	OID	Syntax	Access	Status	Supported
reachDSLSpectrumMgmt AtucMaxTxRate	{ reachDSLSpectrumMgmtLineInfoEntry 1 }	Unsigned32	read-only	deprecated[a]	N
reachDSLSpectrumMgmt AtucMinTxRate	{ reachDSLSpectrumMgmtLineInfoEntry 2 }	Unsigned32	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt AtucMaxTxPower	{ reachDSLSpectrumMgmtLineInfoEntry 3 }	INTEGER(-140…120)	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt AturMaxTxRate	{ reachDSLSpectrumMgmtLineInfoEntry 4 }	Unsigned32	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt AturMinTxRate	{ reachDSLSpectrumMgmtLineInfoEntry 5 }	Unsigned32	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt AturMaxTxPower	{ reachDSLSpectrumMgmtLineInfoEntry 6 }	INTEGER(-140…120)	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt MinEWL	{ reachDSLSpectrumMgmtLineInfoEntry 7 }	Unsigned32	read-only	deprecated[a.]	N
reachDSLSpectrumMgmt MaxEWL	{ reachDSLSpectrumMgmtLineInfoEntry 8 }	Unsigned32	read-only	deprecated[a.]	N

a. The object's status states "current," but there is a note in the description that states that the object is deprecated.

5.38.4.4.1 reachDSLSpectrumMgmtAtucMaxTxRate

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This rate is the max allowed Tx Rate (in bps) for the port based on the Spectrum Management requirements for ATUC in a particular spectral compliance configuration.

5.38.4.4.2 reachDSLSpectrumMgmtAtucMinTxRate

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This rate is the minimum allowed Tx Rate based on the spectrum management requirements for the ATUC. The MinTxRate upper boundary is the "reachDSLSpectrumMgmtAtucMaxTxRate" object.

5.38.4.4.3 reachDSLSpectrumMgmtAtucMaxTxPower

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The maximum transmit power at the ATUC (in tenths of a decibel). This value is based on the spectrum management selection.

5.38.4.4.4 reachDSLSpectrumMgmtAturMaxTxRate

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This rate is the max allowed Tx Rate (in bps) for the port based on the Spectrum Management requirements for ATUC in a particular spectral compliance configuration.

5.38.4.4.5 reachDSLSpectrumMgmtAturMinTxRate

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This rate is the minimum allowed Tx Rate based on the spectrum management requirements for the ATUR. The MinTxRate upper boundary is the “reachDSLSpectrumMgmtAturMaxTxRate” object.

5.38.4.4.6 reachDSLSpectrumMgmtAturMaxTxPower

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The maximum transmit power at the ATUR (in tenths of a decibel). This value is based on the spectrum management selection.

5.38.4.4.7 reachDSLSpectrumMgmtMinEWL

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The minimum value (in feet, if “EWLUnits” is “feet,” or in meters, if “EWLUnits” is “meters”) that the “ConfEWL” object can accept. Together with the “MaxEWL” object, this object indicates the range of EWL when “EWLUnits” is either “feet” or “meters.”

Note: This object is valid only if the “LoopMeasurementMethod” object is “ewl.”

5.38.4.4.8 reachDSLSpectrumMgmtMaxEWL

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The maximum value (in feet, if “EWLUnits” is “feet,” or in meters, if “EWLUnits” is “meters”) that the “ConfEWL” object can accept. Together with the “MinEWL” object, this object indicates the range of EWL when “EWLUnits” is either “feet” or “meters.”

Note: This object is valid only if the “LoopMeasurementMethod” object is “ewl.”

5.38.4.5 reachDSLLineTable

The ReachDSL Line Table.

Note: This entry can be accessed only for ifIndex values that belong to a physical interface, for example, for ifType = “adsl.”

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- reachDSLLineTable - { pdnReachDSLObjects 5 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1.5
- reachDSLLineEntry - { reachDSLLineTable 1 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.20.1.5.1

Table 326. reachDSLLineTable

Object	OID	Syntax	Access	Status	Supported
reachDSLPotsDetectionVoltage	{ reachDSLLineEntry 1 }	INTEGER (0…74)	read-only[a]	current	Y

Table 326. reachDSLLineTable

Object	OID	Syntax	Access	Status	Supported
reachDSLCircuitIdentifier	{ reachDSLLineEntry 2 }	DisplayString (SIZE(1…32))	read-write	deprecated	N

a. Product imposed limit

5.38.4.5.1 reachDSLPotsDetectionVoltage

This object defines the configurable voltage, in units of volts, used to detect the presence of POTS on the DSL line. The range is 0-74 volts. The MIB recommended default value is 3 volts.

Note: This object can be accessed only for ifIndex values that belong to a physical interface. That is, ifType = adsl/ reachDSL.

Valid Object Value(s)

This object is restricted to read-only for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM. The value it returns is based on parameters set in the Initialization Configuration File.

5.38.4.5.2 reachDSLCircuitIdentifier

The value of this object reflects the transmission vendor's circuit identifier to facilitate troubleshooting.

Note: This object can be accessed only for ifIndex values that belong to a physical interface, for example, for ifType = “adsl.”

This object is deprecated in favor of ifAlias. It is highly recommended that future implementations use ifAlias to store the Circuit Identifier associated with reach DSL interfaces.

5.38.4.6 reachDSLSpectrumMgmtLoopMeasurementMethod

The following are valid values for this object:

- none(1) - Indicates that the Spectral Compliance is NOT dependent on any Loop Length measurements or configurations.
- loopLength(2) - The “ConfLoopLength” object is used to determine spectral compliance. The “ConfEWL” & “ConfQuadMode” objects are not used in this mode.
- ewl(3) - The “ConfEWL” object is used to configure the estimated loop length, which is used to determine spectral compliance. In this mode the “ConfLoopLength” & “ConfQuadMode” objects are not used.
- quadMode(4) - The “ConfQuadMode” object is used to determine spectral compliance. The “ConfEWL” & “ConfLoopLength” objects are not used in this mode.

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This object determines which of the three objects is used (if any) to enforce spectrally compliant maximum “Tx Rates” and the maximum allowed “Tx Power” on a given circuit:

- ConfEWL
- ConfLoopLength
- ConfQuadMode

5.38.4.7 reachDSLSpectrumMgmtEWLUnits

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

The following are valid values for this object:

- none(1) - Indicates that the Spectral Compliance is NOT dependent on the “ConfEWL” object
- feet(2 - The “ConfEWL” is measured and configured in feet
- meters(3) - The “ConfEWL” is measured and configured in meters

This object indicates the units of measurement used in describing the Estimated Working Length of a circuit. Following are the details of each of the units:

5.38.4.8 reachDSLSpectrumMgmtMode

This object has been deprecated. The corresponding object in the new pdn_spectrumMgmt.mib should be used instead.

This object restricts the MAX-ACCESS of the “Selection” object. Following are the details of each of the values:

- enableOnly(1) - Indicates that the “Selection” object IS always “enabled” and cannot be set to “disabled”
- disableOnly(2) - Indicates that the “Selection” object IS always “disabled” and cannot be set to “enabled”
- both(3) - Indicates that the “Selection” object will accept both the “enabled” and “disabled” values

5.38.5 Conformance Information

5.38.5.1 Compliance

5.38.5.2 pdnReachDSLCompliance

The compliance statement for SNMP entities which manage ReachDSL interfaces.

Mandatory Groups:

- pdnReachDSLGeneralConfigGroup
- pdnReachDSLGeneralInformationGroup

Optional Groups:

- pdnReachDSLEWLModeGroup - This group is mandatory for reachDSL interfaces whose loop/line length measurement method is EWL, that is, the reachDSLSpectrumMgmtLoopMeasurementMethod object is “ewl.”
- pdnReachDSLLoopLengthModeGroup - This group is mandatory for reachDSL interfaces whose loop/line length measurement method is LoopLength (short / medium/long), that is, the reachDSLSpectrumMgmtLoopMeasurementMethod object is loopLength.”
- pdnReachDSLQuadModeGroup - This group is mandatory for reachDSL interfaces whose loop/line length measurement method is based on Quads, that is, the reachDSLSpectrumMgmtLoopMeasurementMethod object is “quadMode.”

Objects that have a MIN-ACCESS defined:

- reachDSLSpectrumMgmtSelection - MIN-ACCESS read-only
  Write access is not required.
- reachDSLSpectrumMgmtConfAtucMaxTxPower-MIN-ACCESS read-only
  Write access is not required.
- reachDSLSpectrumMgmtConfAturMaxTxPower-MIN-ACCESS read-only
  Write access is not required.
- reachDSLPotsDetectionVoltage - MIN-ACCESS read-only
  Write access is not required.

5.38.6 Conformance

Table 327. PDN-REACHDSL-MIB Conformance

Group	Description	Objects	Supported
pdnReachDSLGeneralConfigGroup	A collection of configuration objects required for providing access to configure all reachDSL interfaces.	reachDSLSpectrumMgmtSelection	N
		reachDSLSpectrumMgmtConfAtucMaxTxPower	Y
		reachDSLSpectrumMgmtConfAturMaxTxPower	Y
		reachDSLPotsDetectionVoltage	Y
pdnReachDSLGeneralInformationGroup	A collection of objects providing information applicable to all reachDSL interfaces.	reachDSLSpectrumMgmtAtucMaxTxRate	N
		reachDSLSpectrumMgmtAtucMinTxRate	N
		reachDSLSpectrumMgmtAtucMaxTxPower	N
		reachDSLSpectrumMgmtAturMaxTxRate	N
		reachDSLSpectrumMgmtAturMinTxRate	N
		reachDSLSpectrumMgmtAturMaxTxPower	N
		reachDSLSpectrumMgmtLoopMeasurementMethod	N
		reachDSLSpectrumMgmtEWLUnits	N
		reachDSLSpectrumMgmtMode	N
pdnReachDSLEWLModeGroup [a]	A collection of objects providing configuration access and information specific to reachDSL interfaces whose loop/line length measurement method is EWL.	reachDSLSpectrumMgmtConfEWL	N
		reachDSLSpectrumMgmtMinEWL	N
		reachDSLSpectrumMgmtMaxEWL	N
pdnReachDSLLoopLengthModeGroup [a]	A collection of object(s) providing configuration access and information specific to reachDSL interfaces whose loop/line length measurement method is Loop Length (short/medium/long); that is, the reachDSLSpectrumMgmt-LoopMeasurementMethod object is “loopLength.”	reachDSLSpectrumMgmtConfLoopLength	N
pdnReachDSLQuadModeGroup [a]	A collection of object(s) providing configuration access and information specific to reachDSL interfaces whose loop/line length measurement method is based on the Quad configuration, that is, the reachDSLSpectrumMgmt-LoopMeasurementMethod object is “quadMode.”	reachDSLSpectrumMgmtConfQuadMode	N

a. See Mutual Exclusion Note below.

Mutual Exclusion Note

The groups below are mutually exclusive; at most one of these groups is mutually exclusive; at most one of these groups is implemented for a particular reachDSL interface. If the Loop Measurement Method is “none,” then none of these groups should be implemented.implemented for a particular reachDSL interface.

- pdnReachDSLEWLModeGroup
- pdnReachDSLLoopLengthModeGroup
- pdnReachDSLQuadModeGroup

5.39 PDN-SECURITY-MIB — pdn_Security.mib

5.39.1 Introduction

This group is in addition to the community name and provides an additional level of security not provided by the community name in version 1 of the SNMP standard.

The version of the MIB used for this document had a comment stating that it was updated on May 21, 2002.

5.39.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.39.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2
- pdn-security - { pdn-common 8 } 1.3.6.1.4.1.1795.2.24.2.8

5.39.4 MIB Objects

Table 328. PDN-SECURITY-MIB Objects

Object	OID	Syntax	Access	Status	Supported
devSecurityMgrValidation	{ pdn-security 1 }	INTEGER disable (1) enable (2)	read-write	mandatory	Y
devSecurityMgrMaxNumber	{ pdn-security 2 }	INTEGER	read-only	mandatory	N
devSecurityMgrCurrentNumber	{ pdn-security 3 }	INTEGER	read-only	mandatory	N
devSecurityMgrTable	{ pdn-security 4 }	SEQUENCE OF devSecurityMgrEntry	not-accessible	deprecated	N
newSecurityMgrTable	{ pdn-security 5 }	SEQUENCE OF newSecurityMgrEntry	not-accessible	deprecated	N
devSecurityTelnetSourceValidation	{ pdn-security 6 }	INTEGER disable(1) enable(2)	read-write	mandatory	N
devSecurityFtpSourceValidation	{ pdn-security 7 }	INTEGER disable(1) enable(2)	read-write	mandatory	N
securityMgrTable	{ pdn-security 8 }	SEQUENCE OF securityMgrEntry	not-accessible	mandatory	Y
devSecuritySNMPMgrAccess	{ pdn-security 9 }	INTEGER disable(1) enable(2)	read-write	mandatory	Y

5.39.4.1 devSecurityMgrValidation

This object is used to either enable or disable SNMP IP validation. The following values are defined for this object:

- disable (1)
- enable (2)

When enabled the first check on any incoming PDU is based on the IP address of the datagram. If the IP address is configured in the devSecurityMgrTable then the community name is checked. If the community name is correct then the lowest access level between the Table and the community name is taken.

5.39.4.2 devSecurityMgrMaxNumber

Maximum number of managers allowed in the devSecurityMgrTable.

5.39.4.3 devSecurityMgrCurrentNumber

Current number of managers in the devSecurityMgrTable.

5.39.4.4 devSecurityMgrTable

Deprecated: 10/7/96

The devSecurityMgrTable has been replaced with the newSecurityMgrTable.

5.39.4.5 newSecurityMgrTable

Deprecated: 5/21/02

5.39.4.6 devSecurityTelnetSourceValidation

This object is used to either enable or disable telnet source address screening. The following are valid values:

- disable(1)
- enable(2)

5.39.4.7 devSecurityFtpSourceValidation

This object is used to either enable or disable ftp source address screening. The following values are defined for this object:

- disable(1)
- enable(2)

5.39.4.8 securityMgrTable

A list of the security manager entries. If an entry is not found in this table, it is added to the table.

Indices:

- securityMgrIpAddress
- securityMgrSubnetMask

OIDs:

- securityMgrTable - { pdn-security 8 } 1.3.6.1.4.1.1795.2.24.2.8
- securityMgrEntry - { securityMgrTable 1 } 1.3.6.1.4.1.1795.2.24.2.8.1

Table 329. securityMgrTable

Object	OID	Syntax	Access	Status	Supported
securityMgrIpAddress	{ securityMgrEntry 1 }	IpAddress	read-only	mandatory	Y
securityMgrSubnetMask	{ securityMgrEntry 2 }	IpAddress	read-only	mandatory	Y

Table 329. securityMgrTable

Object	OID	Syntax	Access	Status	Supported
securityMgrSnmpAccess	{ securityMgrEntry 3 }	INTEGER readOnly(1) readWrite(2) noAccess(3)	read-write	mandatory	Y
securityMgrTelnetAccess	{ securityMgrEntry 4 }	INTEGER disable(1) enable(2)	read-write	mandatory	N
securityMgrFtpAccess	{ securityMgrEntry 5 }	INTEGER disable(1) enable(2)	read-write	mandatory	N
securityMgrTrapAccess	{ securityMgrEntry 6 }	INTEGER trapAccess(1) noTrapAccess(2)	read-write	mandatory	Y
securityMgrRowStatus	{ securityMgrEntry 7 }	RowStatus	read-write	mandatory	Y

5.39.4.8.1 securityMgrIpAddress

This is used to specify the IP address that identifies the SNMP manager(s) that are authorized to send SNMP messages.

5.39.4.8.2 securityMgrSubnetMask

This object is used to specify the subnet mask

Valid Object Value(s)

This object is required but ignored for all operations. The implied mask is 255.255.255.255. As long as some mask value is present, the SNMP operation will succeed. During get type operations, the mask is returned as 255.255.255.255.

5.39.4.8.3 securityMgrSnmpAccess

This is used to specify the snmp access to the device. The following values are defined for this object:

- readOnly(1)
- readWrite(2)
- noAccess(3)

Valid Object Value(s)

The following valid object values are supported:

- readWrite(2) - Indicate that the securityMgrIpAddress maps to a SNMP Manager
- noAccess(3) - Indicates that the securityMgrIpAddress does not map to a SNMP Manager

5.39.4.8.4 securityMgrTelnetAccess

This is used to specify the telnet access to the device. The following are valid values:

- disable(1)
- enable(2)

5.39.4.8.5 securityMgrFtpAccess

This is used to specify the ftp access to the device. The following values are defined for this object:

- disable(1)

- enable(2)

5.39.4.8.6 securityMgrTrapAccess

This is used to specify the trap access to the device. The following values are defined for this object:

- trapAccess(1)
- noTrapAccess(2)

5.39.4.8.7 securityMgrRowStatus

The status of this table entry.

5.39.4.9 devSecuritySNMPMgrAccess

This object is used to either enable or disable SNMP access to the system. When disabled the first check on any incoming PDU is based on the UPD type of the datagram. If it is an SNMP manager trying to access SNMP agent on the system the access is denied.

5.40 PDN-SPECTRUMMGMT-MIB — pdn_spectrummgmt.mib

5.40.1 Introduction

This MIB is written to facilitate spectrum management of any xDSL interface. The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of "200501190000Z," or January 19, 2005.

5.40.2 Textual Conventions

No "special" textual conventions are made or referenced in this MIB.

5.40.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2
- pdn-interfaces - { pdn-common 6 } 1.3.6.1.4.1.1795.2.24.2.6

- pdnSpectrumMgmt - { pdn-interfaces 19 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.19
- pdnSpecMgmtObjects - { pdnSpectrumMgmt 1 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1
- pdnNewSpecMgmtObjects - { pdnSpectrumMgmt 2 } 1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2

5.40.4 MIB Objects

MIB objects under the pdnSpecMgmtObjects branch have been deprecated and are not shown.

Table 330. PDN-SPECTRUMMGMT-MIB Objects

Object	OID	Syntax	Access	Status	Supported
newSpectrumMgmt GeneralConfigTable	{ pdnNewSpecMgmtObjects 1 }	SEQUENCE of newSpectrumMgmtGeneral-ConfigEntry	not-accessible	current	Y
newSpectrumMgmt ConfTable	{ pdnNewSpecMgmtObjects 2 }	SEQUENCE of newSpectrumMgmtConfEntry	not-accessible	current	Y
newSpectrumMgmt LineInfoTable	{ pdnNewSpecMgmtObjects 3 }	SEQUENCE of newSpectrumMgmtLineInfoEntry	not-accessible	current	Y

5.40.4.1 newSpectrumMgmtGeneralConfigTable

General Configuration of the spectrum management. This table contains an entry for each technology managed by this device.

Indicies:

- entPhysicalIndex (Section 5.6.5.1.1.1, Page 87)
- ifType (Section 5.3.4.3.3, Page 41)

OIDs:

- newSpectrumMgmtGeneralConfigTable - { pdnNewSpecMgmtObjects 1 }
  1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2.1
- newSpectrumMgmtGeneralConfigEntry - { newSpectrumMgmtGeneralConfigTable 1 }
  1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2.1.1

Table 331. newSpectrumMgmtGeneralConfigTable

Object	OID	Syntax	Access	Status	Supported
newSpectrumMgmtSelection	{ newSpectrumMgmtGeneralConfigEntry 1 }	INTEGER enabled(1) disabled(2)	read-write	current	Y
newSpectrumMgmtMode	{ newSpectrumMgmtGeneralConfigEntry 2 }	INTEGER enableOnly(1) disableOnly(2) both(3)	read-only	current	Y
newSpectrumMgmtLoop-MeasurementMethod	{ newSpectrumMgmtGeneralConfigEntry 3 }	INTEGER none(1) loopLength(2) ewl(3), quadMode(4)	read-only	current	Y
newSpectrumMgmt-EWLUnits	{ newSpectrumMgmtGeneralConfigEntry 4 }	INTEGER none(1) feet(2) meters(3)	read-only	current	Y
newSpectrumMgmtStd	{ newSpectrumMgmtGeneralConfigEntry 5 }	INTEGER none(1) other(2) ansiT1417(3) anfpND1602(4) jj10001(5)	read-write	current	Y

5.40.4.1.1 newSpectrumMgmtSelection

The following values are defined for this object:

- enabled(1)
- disabled(2)

This object is used to Enable or Disable Spectrum Management on DSL interfaces managed by this device. Enabling this object limits the DSL speeds on each port based on line length in order to meet ANSI T1.417 Spectrum Management requirements or BT Access Network Spectrum requirements.

The values that this object can accept are controlled by the “Mode” object.

In cases where a particular spectrum management plan or standards needs to be selected, the newSpectrumMgmtStd (Section 5.40.4.1.5, Page 475) can be used to make that selection.

5.40.4.1.2 newSpectrumMgmtMode

This object restricts the MAX-ACCESS of the “Selection” object. The following values are defined for this object:

- enableOnly(1) - Indicates that the “Selection” object IS always “enabled” and cannot be set to “disabled”.
- disableOnly(2) - Indicates that the “Selection” object IS always “disabled” and cannot be set to “enabled”.
- both(3) - Indicates that the “Selection” object will accept both the “enabled” and “disabled” values.

5.40.4.1.3 newSpectrumMgmtLoopMeasurementMethod

This object indicates the measurement method used in describing the Loop Length of a circuit. Following are the details of each of the options:

- none(1) - Indicates that the Spectral Compliance is NOT dependent on any Loop Length measurements or configurations.
- loopLength(2) - The “ConfLoopLength” object is used to determine spectral compliance. The “ConfEWL” & “ConfQuadMode” objects are not used in this mode.
- ewl(3) - The “ConfEWL” object is used to configure the estimated loop length, which is used to determine spectral compliance. In this mode the “ConfLoopLength” & “ConfQuadMode” objects are not used.
- quadMode(4) - The “ConfQuadMode” object is used to determine spectral compliance. The “ConfEWL” & “ConfLoopLength” objects are not used in this mode.

This object determines which of the three following objects (if any) is used to spectrally compliant maximum “Tx Rates” and the maximum allowed “Tx Power” on a given circuit:

- ConfEWL
- ConfLoopLengt
- ConfQuadMode

5.40.4.1.4 newSpectrumMgmtEWLUnits

This object indicates the units of measurement used in describing the Estimated Working Length of a circuit. Following are the details of each of the units:

- none(1) - Indicates that the Spectral Compliance is NOT dependent on the “ConfEWL” object
- feet(2) - The “ConfEWL” is measured and configured in feet
- meters(3) - The “ConfEWL” is measured and configured in meters

5.40.4.1.5 newSpectrumMgmtStd

This object indicates the spectrum management plan or standard to which the entry is complying with. The following are valid values:

- none(1) - No special compliance scheme is specified.
- other(2) - None of the selections listed next. An example of other(2) could be the case where initialization configuration files are used to specify spectrum management compliance parameters.
- ansiT1417(3) - Alliance for Telecommunications Industry Solutions T1.417 Spectrum Management for Loop Transmission Systems.
- anfpND1602(4) - Access Network Frequency Plan as specified n the Network Interoperability Consultative Committee’s ND1602 document.
- jj10001(5) - Telecommunications Technology Committee (TTC) standard, “Spectrum Management for Metallic Trans mission Systems Version 1.”

5.40.4.2 newSpectrumMgmtConfTable

The new Spectrum Management Configuration Table This table is used for configuring spectrum management objects on any DSL interface.

Index:

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- newSpectrumMgmtConfTable - { pdnNewSpecMgmtObjects 2 }

1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2.2

- newSpectrumMgmtConfEntry - { newSpectrumMgmtConfTable 1 }

1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2.2.1

Table 332. newSpectrumMgmtConfTable

Object	OID	Syntax	Access	Status	Supported
newSpectrumMgmtConf EWL	{ newSpectrumMgmtConfEntry 1 }	Unsigned32	read-write	current	Y
newSpectrumMgmtConf LoopLength	{ newSpectrumMgmtConfEntry 2 }	INTEGER short(1) medium(2) long(3) extraShort(4)	read-write	current	Y
newSpectrumMgmtConf QuadMode	{ newSpectrumMgmtConfEntry 3 }	INTEGER sameQuad(1) segregatedQuadUpto3km(2) segregatedQuadAbove3km(3)	read-write	current	Y

5.40.4.2.1 newSpectrumMgmtConfEWL

The Equivalent Working Length (EWL). The value selected corresponds to the estimated length of the line in:

- feet - if “EWLUnits” object is “feet”
- meters - if “EWLUnits” object is “meters”

This value is usually specified by the carrier and MUST be in integral multiples of:

- 500 feet, if “EWLUnits” is “feet”
- 200 meters, if “EWLUnits” is “meters”

The range of values that this object can take is determined by the “MinEWL” and “MaxEWL” objects. Changing this object may limit the maximum transmit power and the effective DSL speeds at which the ports train up.

Note: This object is used only if Spectrum Management is enabled and “LoopMeasurementMethod” is “ewl.” This object shares a mutual exclusion relationship with the “ConfLoopLength” & “ConfQuadMode” objects, as only one of the two is used for configuration at any time.

5.40.4.2.2 newSpectrumMgmtConfLoopLength

The following values are defined for this object:

- short(1)
- medium(2)
- long(3)

- extraShort(4)

This entry specifies the loop length for the line. The configurable values are short(1), medium(2) and long(3). The "Loop Length" feature can be used only if the "shdslSpectrumMgmtSelection" object is Enabled and the "LoopMeasurementMethod" is "loopLength." Changing this object may limit the maximum transmit power and the effective DSL speeds at which ports train up.

Note: This object shares a mutual exclusion relationship with the "ConfEWL" objects, that is, only one of the three can be used for configuration at one time.

5.40.4.2.3 newSpectrumMgmtConfQuadMode

The following values are defined for this object:

- sameQuad(1)
- segregatedQuadUpto3km(2)
- segregatedQuadAbove3km(3)

This entry specifies the quad cable configuration & length for the line. The configurable values are sameQuad(1), segregatedQuadUpto3km(2) and segregatedQuadAbove3km(3). The "Quad Mode" feature can be used only if the "reachDSLSpectrumMgmtSelection" object is Enabled and the "LoopMeasurementMethod" is "quadMode." Changing this object may limit the maximum transmit power and the effective DSL speeds at which ports train up.

Note: This object shares a mutual exclusion relationship with the "ConfEWL" & "ConfLoopLength" objects, that is, only one of the three can be used for configuration at one time.

5.40.4.3 newSpectrumMgmtLineInfoTable

The Spectrum Management Line Info Table. This table gives information about the spectrum management ranges that are applicable for each port, based on the spectral requirements for the device.

Index

- ifIndex (Section 5.3.4.3.1, Page 37)

OIDs:

- newSpectrumMgmtLineInfoTable - { pdnNewSpecMgmtObjects 3 }

1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2.3

- newSpectrumMgmtLineInfoEntry - { newSpectrumMgmtLineInfoTable 1 }

1.3.6.1.4.1.1795.2.24.2.24.2.6.19.1.2.3.1

Table 333. newSpectrumMgmtLineInfoTable (1 of 2)

Object	OID	Syntax	Access	Status	Supported
newSpectrumMgmt XtucMax1TxRate	{ newSpectrumMgmtLineInfoEntry 1 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XtucMin1TxRate	{ newSpectrumMgmtLineInfoEntry 2 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XtucMax2TxRate	{ newSpectrumMgmtLineInfoEntry 3 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XtucMin2TxRate	{ newSpectrumMgmtLineInfoEntry 4 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XtucMaxTxPower	{ newSpectrumMgmtLineInfoEntry 5 }	Integer32(-140…120)	read-only	current	Y
newSpectrumMgmt XturMax1TxRate	{ newSpectrumMgmtLineInfoEntry 6 }	Unsigned32	read-only	current	Y

Table 333. newSpectrumMgmtLineInfoTable (2 of 2)

Object	OID	Syntax	Access	Status	Supported
newSpectrumMgmt XturMin1TxRate	{ newSpectrumMgmtLineInfoEntry 7 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XturMax2TxRate	{ newSpectrumMgmtLineInfoEntry 8 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XturMin2TxRate	{ newSpectrumMgmtLineInfoEntry 9 }	Unsigned32	read-only	current	Y
newSpectrumMgmt XturMaxTxPower	{ newSpectrumMgmtLineInfoEntry 10 }	Integer32(-140…120)	read-only	current	Y
newSpectrumMgmt MinEWL	{ newSpectrumMgmtLineInfoEntry 11 }	Unsigned32	read-only	current	Y
newSpectrumMgmt MaxEWL	{ newSpectrumMgmtLineInfoEntry 12 }	Unsigned32	read-only	current	Y

5.40.4.3.1 newSpectrumMgmtXtucMax1TxRate

This rate is the max allowed Tx Rate (in bps) for the port based on the Spectrum Management requirements for XTUC in a particular spectral compliance configuration.

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.2 newSpectrumMgmtXtucMin1TxRate

This rate is the min allowed Tx Rate based on the spectrum management requirements for the XTUC. The MinTxRate upper boundary is the “newSpectrumMgmtXtucMax1TxRate” object.

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.3 newSpectrumMgmtXtucMax2TxRate

This rate is the max allowed Tx Rate (in bps) for the port based on the Spectrum Management requirements for XTUC in a particular spectral compliance configuration. This object is the

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.4 newSpectrumMgmtXtucMin2TxRate

This rate is the min allowed Tx Rate based on the spectrum management requirements for the XTUC. The MinTxRate upper boundary is the “newSpectrumMgmtXtucMax1TxRate” object.

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.5 newSpectrumMgmtXtucMaxTxPower

The maximum transmit power at the ATUC (in tenths of a decibel). This value is based on the spectrum management selection.

5.40.4.3.6 newSpectrumMgmtXturMax1TxRate

This rate is the max allowed Tx Rate (in bps) for the port based on the Spectrum Management requirements for XTUR in a particular spectral compliance configuration. This object is the

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.7 newSpectrumMgmtXturMin1TxRate

This rate is the min allowed Tx Rate based on the spectrum management requirements for the XTUR. The MinTxRate upper boundary is the “newSpectrumMgmtXturMax1TxRate” object.

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.8 newSpectrumMgmtXturMax2TxRate

This rate is the max allowed Tx Rate (in bps) for the port based on the Spectrum Management requirements for XTUR in a particular spectral compliance configuration.

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.9 newSpectrumMgmtXturMin2TxRate

This rate is the min allowed Tx Rate based on the spectrum management requirements for the XTUR. The MinTxRate upper boundary is the “newSpectrumMgmtXtucMax1TxRate” object.

Some measurement methods may have more than one range for the TxRate. Min1TxRate and Max1TxRate will always have the first range of values and the Min2TxRate and Max2TxRate will have the second range if the method demands it.

5.40.4.3.10 newSpectrumMgmtXturMaxTxPower

The maximum transmit power at the ATUR (in tenths of a decibel). This value is based on the spectrum management selection.

5.40.4.3.11 newSpectrumMgmtMinEWL

The minimum value (in feet, if “EWLUnits” is “feet, or in meters, if “EWLUnits” is “meters”) that the “ConfEWL” object can accept. Together with the “MaxEWL” object, this object indicates the range of EWL when “EWLUnits” is either “feet” or “meters.”

Note: This object is valid only if the “LoopMeasurementMethod” object is “ewl.”

5.40.4.3.12 newSpectrumMgmtMaxEWL

The maximum value (in feet, if “EWLUnits” is “feet,” or in meters, if “EWLUnits” is “meters”) that the “ConfEWL” object can accept. Together with the “MinEWL” object, this object indicates the range of EWL when “EWLUnits” is either “feet” or “meters.”

Note: This object is valid only if the “LoopMeasurementMethod” object is “ewl.”

5.40.5 Conformance Information

5.40.6 Compliance

5.40.6.1 pdnSpecMgmtCompliance

The compliance statement for SNMP entities that control spectrum management on interfaces.

Mandatory Groups:

- pdnGeneralConfigGroup
- pdnLineInfoGroup

Optional Groups:

Note:These groups are mutually exclusive; that is, at most one of these groups is implemented for a particular DSL interface. If the Loop Measurement Method is "none," then none of these groups should be implemented.

- pdnEWLModeGroup - This group is mandatory for DSL interfaces whose loop/line length measurement method is EWL, that is, the newSpectrumMgmtLoopMeasurementMethod object is "ewl."
- pdnLoopLengthModeGroup - This group is mandatory for DSL interfaces whose loop/line length measurement method is LoopLength, that is, the newSpectrumMgmtLoopMeasurementMethod object is "loopLength."
- pdnQuadModeGroup - This group is mandatory for DSL interfaces whose loop/line length measurement method is based on Quads, that is, the newSpectrumMgmtLoopMeasurementMethod object is "quadMode."
- pdnSpectrumMgmtStdGroup - This group is mandatory for DSL interfaces that need to be configured to comply with a particular spectrum management plan or standard.

Objects with a MIN-ACCES defined:

- newSpectrumMgmtSelection - MIN-ACCESS read-only
  Write access is not required.

5.40.7 Conformance

Table 334. PDN-SPECTRUMMGMT-MIB Conformance (1 of 2)

Group	Description	Objects	Supported
pdnGeneralConfigGroup	A collection of general configuration objects required for the spectrum management of interfaces.	newSpectrumMgmtSelection	Y
		newSpectrumMgmtMode	Y
		newSpectrumMgmtLoopMeasurementMethod	Y
		newSpectrumMgmtEWLUnits	Y

Table 334. PDN-SPECTRUMMGMT-MIB Conformance (2 of 2)

Group	Description	Objects	Supported
pdnLineInfoGroup	A collection of objects providing information regarding spectrum management of interfaces.	newSpectrumMgmtXtucMax1TxRate	Y
		newSpectrumMgmtXtucMin1TxRate	Y
		newSpectrumMgmtXtucMax2TxRate	N
		newSpectrumMgmtXtucMin2TxRate	N
		newSpectrumMgmtXtucMaxTxPower	Y
		newSpectrumMgmtXturMax1TxRate	Y
		newSpectrumMgmtXturMin1TxRate	Y
		newSpectrumMgmtXturMax2TxRate	N
		newSpectrumMgmtXturMin2TxRate	N
		newSpectrumMgmtXturMaxTxPower	Y
		newSpectrumMgmtMinEWL	Y
		newSpectrumMgmtMaxEWL	Y
pdnEWLModeGroup	A collection of configuration objects used to control spectrum management of DSL interfaces when the measurement method is “EWL.”	newSpectrumMgmtConfEWL	Y
		newSpectrumMgmtMinEWL	Y
		newSpectrumMgmtMaxEWL	Y
pdnLoopLengthModeGroup	A collection of configuration objects used to control spectrum management of DSL interfaces when the measurement method is “LoopLength.”	newSpectrumMgmtConfLoopLength	Y
pdnQuadModeGroup	A collection of object(s) providing configuration access and information specific to DSL interfaces whose loop/ line length measurement method is based on the Quad configuration, that is, the newSpectrumMgmtLoopMeasurementMethod object is “quadMode.”	newSpectrumMgmtConfQuadMode	Y
pdnSpectrumMgmtStdGroup	A collection of object(s) providing configuration to comply with a particular spectrum management plan or standard.	newSpectrumMgmtStd	Y

5.41 PDN-STACKABLE-MIB — PDN-STACKABLE-MIB.mib

5.41.1 Introduction

This MIB contains objects that are used for configuration and statistics reporting things that are unique to a stackable product. The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of "200309140000Z," or September 14, 2004.

5.41.2 Textual Conventions

No "special" textual conventions are made or referenced in this MIB.

5.41.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnStackable - { pdn-common 36 } 1.3.6.1.4.1.1795.2.24.2.36
- pdnStackableNotifications - { pdnStackable 3 } 1.3.6.1.4.1.1795.2.24.2.36.0
- pdnStackableObjects - { pdnStackable 1 } 1.3.6.1.4.1.1795.2.24.2.36.1
- pdnStackableAFNs - { pdnStackable 2 } 1.3.6.1.4.1.1795.2.24.2.36.2

5.41.4 MIB Objects

Table 335. PDN-STACKABLE-MIB Objects (1 of 2)

Object	OID	Syntax	Access	Status	Supported
wanInterface	{ pdnStackableObjects 1 }	BITS stackLink1(0) stackLink2(1) plugInModule(2) stackLink3(3) stackLink4(4) stackLink5(5) stackLink6(6) stackLink7(7) stackLink8(8) stackLink9(9) stackLink10(10) stackLink11(11) stackLink12(12) stackLink13(13) stackLink14(14) stackLink15(15) stackLink16(16) stackLink17(17) stackLink18(18)	read-write	current	Y

Table 335. PDN-STACKABLE-MIB Objects (2 of 2)

Object	OID	Syntax	Access	Status	Supported
pdnStackMethod	{ pdnStackableObjects 2 }	INTEGER daisyChain(1) star(2)	read-only	current	N
pdnStackConfiguration-Table	{ pdnStackableObjects 3 }	SEQUENCE OF pdnStackConfigurationEntry	not-accessible	current	N
pdnMoveUnitCmdTable	{ pdnStackableObjects 4 }	SEQUENCE OF pdnMoveUnitCmdEntry	not-accessible	current	N
pdnStackUnitAdded-TrapEnable	{ pdnStackableObjects 5 }	SwitchState	read-write	current	N
pdnStackUnitComFailure-TrapEnable	{ pdnStackableObjects 6 }	SwitchState	read-write	current	N
pdnStackUnitCom-RestoredTrapEnable	{ pdnStackableObjects 7 }	SwitchState	read-write	current	N

5.41.4.1 wanInterface

This object allows the WAN interface to be selected. Given that the syntax for this object is BITS [Section 4.1, Page 11], it is intended that the implementation could support multiple-simultaneous uplinks.

Valid Object Value(s)

The valid values for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM are:

Table 336.

GigE MAUI		T1/E1 ML PPP MAUI	
stackLink1(0)	Ethernet Port 1	stackLink1(0)	EthernetPort 1
stackLink2(1)	Ethernet Port 2	stackLink2(1)	Ethernet Port 2
stackLink3(3)	Ethernet Port 3	stackLink3(3)	T1/E1 PPP 1
		stackLink4(4)	T1/E1 PPP 2
		stackLink5(5)	T1/E1 PPP 3
		stackLink6(6)	T1/E1 PPP 4
		stackLink7(7)	T1/E1 PPP 5
		stackLink8(8)	T1/E1 PPP 6
		stackLink9(9)	T1/E1 PPP 7
		stackLink10(10)	T1/E1 PPP 8
		stackLink11(11)	ML PPP 1
		stackLink12(12)	ML PPP 2
		stackLink13(13)	ML PPP 3
		stackLink14(14)	ML PPP 4
		stackLink15(15)	ML PPP 5
		stackLink16(16)	ML PPP 6

Table 336.

GigE MAUI		T1/E1 ML PPP MAUI	
		stackLink17(17)	ML PPP 7
		stackLink18(18)	ML PPP 8

- stackLink1(0) - This value maps to Ethernet Port 1
- stackLink2(1) - This value maps to Ethernet Port 2
- stackLink3(3) - This value maps to Ethernet Port 3

5.41.4.2 pdnStackMethod

This object identifies the method by which units are interconnected.

Valid Object Value(s)

This object always returns daisyChain(1).

5.41.4.3 pdnStackConfigurationTable

This table is used to display the stack configuration. This table is indexed by the value of an externally visible identifier known as the External ID (reference pdnUnitExternalID, Section 5.41.4.3.1, Page 486 for the actual object used as the External ID). In the BitStorm 2600 and GranDSLAM 4200 IP DSLAM, the External ID is visible as a "STACK POSITION" LED. This table is a static table where the number of rows in this table match the maximum number of supported in a stack. For this product, the maximum number of units supported in a stack is eight. However, there are times where the table may contain an additional row. Consider the case illustrated with the following table in which all External IDs have been assigned and a unit has been added to the stack:

External ID (pdnUnitExternalID)	Entity Physical Index (pdnUnitEntPhysicalIndex)	Assigned (pdnUnitAssigned)	GUID (pdnUnitGUID)	Present (pdnUnitPresent)
1	*n*	true(1)	aa-bb-cc-dd-ee-01	true(1)
2	*n*	true(1)	aa-bb-cc-dd-ee-02	true(1)
3	*n*	true(1)	aa-bb-cc-dd-ee-03	true(1)
4	*n*	true(1)	aa-bb-cc-dd-ee-04	true(1)
5	*n*	true(1)	aa-bb-cc-dd-ee-05	true(1)
6	*n*	true(1)	aa-bb-cc-dd-ee-06	true(1)
7	*n*	true(1)	aa-bb-cc-dd-ee-07	false(2)
8	*n*	true(1)	aa-bb-cc-dd-ee-08	false(2)
99[a]	*n*	true(1)	aa-bb-cc-dd-ee-09	true(1)

a. 99 is a special value described below.

In this case, a unit has been attached to the stack but cannot be assigned an External ID because they have all been assigned. As such, an extra row is created dynamically and marked with a value of 99 to indicate this special condition.

At this point, the user would issue a move command (pdnMoveUnitCmd, Section 5.41.4.4.3, Page 488) to move the External ID corresponding to one of the units that are not present along with its configuration data to the extra unit. For example, using the move command to allow the extra unit to take on the External ID and configuration values of the unit currently associated with External ID of 7, the following values would be assigned:

- pdnMoveUnitSrcNumber = 7
- pdnMoveUnitDestNumber = 99
- pdnMoveUnitCmd = move(2)

After the command is executed this table would look like:

External ID (pdnUnitExternalID)	Entity Physical Index (pdnUnitEntPhysicalIndex)	Assigned (pdnUnitAssigned)	GUID (pdnUnitGUID)	Present (pdnUnitPresent)
1	*n*	true(1)	aa-bb-cc-dd-ee-01	true(1)
2	*n*	true(1)	aa-bb-cc-dd-ee-02	true(1)
3	*n*	true(1)	aa-bb-cc-dd-ee-03	true(1)
4	*n*	true(1)	aa-bb-cc-dd-ee-04	true(1)
5	*n*	true(1)	aa-bb-cc-dd-ee-05	true(1)
6	*n*	true(1)	aa-bb-cc-dd-ee-06	true(1)
7	*n*	true(1)	aa-bb-cc-dd-ee-09	true(1)
8	*n*	true(1)	aa-bb-cc-dd-ee-08	false(2)

Index:

- pdnUnitExternalID

OIDs:

- pdnStackConfigurationTable - { pdnStackableObjects 2 } 1.3.6.1.4.1.1795.2.24.2.36.1.2
- pdnStackConfigurationEntry - { pdnStackConfigurationTable 1 } 1.3.6.1.4.1.1795.2.24.2.36.1.2.1

Table 337. pdnStackConfigurationTable

Object	OID	Syntax	Access	Status	Supported
pdnUnitExternalID	{ pdnStackConfigurationEntry 1 }	Unsigned32	not-accessible	current	N
pdnUnitEntPhysicalIndex	{ pdnStackConfigurationEntry 2 }	Integer32 (1…2147483647)	read-only	current	N
pdnUnitAssigned	{ pdnStackConfigurationEntry 3 }	TruthValue	read-write	current	N
pdnUnitGUID	{ pdnStackConfigurationEntry 4 }	DisplayString	read-only	current	N
pdnUnitPresent	{ pdnStackConfigurationEntry 5 }	TruthValue	read-only	current	N

5.41.4.3.1 pdnUnitExternalID

This object is the External ID used to refer to a particular unit. The value of this External ID will typically be indicated on the unit by some means such as a LED.

5.41.4.3.2 pdnUnitEntPhysicalIndex

This object serves to map the table row to an entry into the entPhysicalTable. This object will contain the entPhysicalIndex associated with the unit.

5.41.4.3.3 pdnUnitAssigned

This object indicates if the External ID has been assigned to a particular Globally Unique Identifier (GUID).

Writing a false(2) is a mechanism to delete a unit from a stack. When a unit is deleted from a stack, all data that the Stack knows about the unit is purged.

On read operations:

- true(1) - indicates that the External ID has been assigned to a unit.
- false(2) - indicates that the External ID has not been assigned to a unit

5.41.4.3.4 pdnUnitGUID

This object is the representation of the units' Globally Unique Identifier (GUID). The requirement for this identifier is that is something that makes the unit uniquely identifiable.

Valid Object Value(s)

The BitStorm 2600 and GranDSLAM 4200 IP DSLAM will use the units MAC address as the GUID.

5.41.4.3.5 pdnUnitPresent

This object identifies if the unit is present in the stack. Note that when a unit is removed from a stack, its stack and configuration information is retained.

5.41.4.4 pdnMoveUnitCmdTable

This table contains the objects necessary to support a command to move a unit in a stack. This command only applies when singleManagementEntity is enabled. The table has a single row supporting a command to move a unit in a stack.

Indices:

- pdnMoveUnitSrcNumber
- pdnMoveUnitDestNumber

OIDs:

- pdnMoveUnitCmdTable - { pdnStackableObjects 3 } 1.3.6.1.4.1.1795.2.24.2.36.1.3
- pdnMoveUnitCmdEntry - { pdnMoveUnitCmdTable 1 } 1.3.6.1.4.1.1795.2.24.2.36.1.3.1

Table 338. pdnMoveUnitCmdTable

Object	OID	Syntax	Access	Status	Supported
pdnMoveUnitSrcNumber	{ pdnMoveUnitCmdEntry 1 }	Unsigned32	not-accessible	current	N
pdnMoveUnitDestNumber	{ pdnMoveUnitCmdEntry 2 }	Unsigned32	not-accessible	current	N
pdnMoveUnitCmd	{ pdnMoveUnitCmdEntry 3 }	INTEGER noOp(1) move(2)	read-write	current	N

5.41.4.4.1 pdnMoveUnitSrcNumber

This object is used as a table index and serves as the identifier for the source unit in the move command. This object identifies the unit with its External ID (See pdnUnitExternalID, Section 5.41.4.3.1, Page 486).

5.41.4.4.2 pdnMoveUnitDestNumber

This object is used, as a table index and serves as the identifier for the destination unit in the move command. This object identifies the unit with its External ID (See pdnUnitExternalID, Section 5.41.4.3.1, Page 486).

5.41.4.4.3 pdnMoveUnitCmd

The following values are defined for this object:

- noOp(1)
- move(2)

Writing the value move(2) will cause the stack and configuration related information for the source unit to be moved to the destination unit. This object always returns a noOp(1) on a read operation.

5.41.4.5 pdnStackUnitAddedTrapEnable

Indicates whether the pdnStackUnitAdded trap should be generated. The following values are defined for this object:

- enabled(1) - Generates the trap
- disabled(2) - Does not generate the trap

This object defaults to enable(1).

5.41.4.6 pdnStackUnitComFailureTrapEnable

Indicates whether the pdnStackUnitComFailure trap should be generated. The following values are defined for this object:

- enabled(1) - Generates the trap
- disabled(2) - Does not generate the trap

This object defaults to enable(1).

5.41.4.7 pdnStackUnitComRestoredTrapEnable

Indicates whether the pdnStackUnitComRestored trap should be generated. The following values are defined for this object:

- enabled(1) - Generates the trap
- disabled(2) - Does not generate the trap

This object defaults to enable(1).

5.41.5 Traps

OIDs:

- pdnStackableNotifications - { pdnStackable 3 } 1.3.6.1.4.1.1795.2.24.2.36.0

Table 339. PDN-STACKABLE-MIB Traps

Notification	OID	Variable Bindings / Objects	Status	Supported
pdnStackUnitAdded	{ pdnStackableNotifications 1 }	pdnUnitGUID	current	N
pdnStackUnitComFailure	{ pdnStackableNotifications 2 }	pdnUnitGUID	current	N
pdnStackUnitComRestored	{ pdnStackableNotifications 3 }	pdnUnitGUID	current	N

5.41.5.1 pdnStackUnitAdded

This trap signifies that a new unit has been added to a stack.

5.41.5.2 pdnStackUnitComFailure

This trap signifies that a loss of communication has occurred with a unit that was part of a stack.

5.41.5.3 pdnStackUnitComRestored

This trap signifies that communication to a unit that was part of the stack has occurred.

5.41.6 Conformance Information

5.41.6.1 Compliance

5.41.6.1.1 pdnStackable Compliance

The compliance statement for elements implementing stackable units.

Mandatory Groups:

No mandatory groups are specified.

Optional Groups:

- wanInterfaceGroup
  This group is only required for implementations that need to select the WAN interface(s).
- singleManagementEntityGroup
  This group is only required for implementations that implement a “single logical management entity. “
- singleManagementEntityNotificationGroup
  This group is only required for implementations that generate notification related to the “single logical management entity.”

5.41.6.2 Conformance

Table 340. PDN-STACKABLE-MIB Conformance

Group	Description	Objects	Supported
wanInterfaceGroup	Objects related to the WAN.	wanInterface	Y
singleManagementEntityGroup	Objects related to the Single Management Entity.	pdnStackMethod	N
		pdnUnitEntPhysicalIndex	N
		pdnUnitAssigned	N
		pdnUnitGUID	N
		pdnUnitPresent	N
		pdnMoveUnitCmd	N
		pdnStackUnitAddedTrapEnable	N
		pdnStackUnitComFailureTrapEnable	N
		pdnStackUnitComRestoredTrapEnable	N
singleManagementEntity-NotificationGroup	Notifications that must be implemented for Single Management Entity implementations.	pdnStackUnitAdded	N
		pdnStackUnitComFailure	N
		pdnStackUnitComRestored	N

5.42 PDN-SYSLOG-MIB — pdn_syslog.mib

5.42.1 Introduction

This MIB contains objects relating to sys log.

The version of the MIB used for this document had a LAST-UPDATED of 0105120000Z which decodes to December 5, 2001.

5.42.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.42.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2
- pdn-syslog - { pdn-common 31 } 1.3.6.1.4.1.1795.2.24.2.31
- pdnSyslog - { pdn-syslog 1 } 1.3.6.1.4.1.1795.2.24.2.31.1

5.42.4 MIB Objects

Table 341. PDN-SYSLOG-MIB Objects (1 of 2)

Object	OID	Syntax	Access	Status	Supported
pdnSyslogStatus	{ pdnSyslog 1 }	INTEGER disable (1) enable (2)	read-write	current	N
pdnSyslogIPAddr	{ pdnSyslog 2 }	IpAddress	read-write	current	Y
pdnSyslogLevel	{ pdnSyslog 3 }	INTEGER emerg(1) err(2) norm(3) info(4)	read-write	deprecated	N
pdnSyslogPort	{ pdnSyslog 4 }	Integer32	read-write	deprecated	Y
pdnSyslogSeverityThreshold	{ pdnSyslog 5 }	INTEGER emerg(0) alert(1) critical(2) error(3) warning(4) notice(5) info(6) debug(7)	read-write	current	Y

Table 341. PDN-SYSLOG-MIB Objects (2 of 2)

Object	OID	Syntax	Access	Status	Supported
pdnSyslogRemoteDaemon	{ pdnSyslog 6 }	INTEGER disable (1) enable (2)	read-write	current	Y
pdnSyslogTable	{ pdnSyslog 7 }	SEQUENCE OF pdnSyslogEntry	not-accessible	current	Y
pdnSyslogNumOfMsgInTable	{ pdnSyslog 8 }	Integer32	read-only	current	Y
pdnSyslogMaxTableSize	{ pdnSyslog 9 }	Integer32	read-only	current	Y
pdnSyslogClearTable	{ pdnSyslog 10 }	INTEGER noOp(1) clear(2)	read-write	current	Y
pdnSyslogMsgToConsole	{ pdnSyslog 11 }	INTEGER disable (1) enable (2)	read-write	current	Y
pdnSyslogRateLimiting	{ pdnSyslog 12 }	INTEGER disable (1) enable (2)	read-write	current	Y

5.42.4.1 pdnSyslogStatus

This value allows the user to disable or enable syslog event generation. The following values are defined for this object:

- disable(1)
- enable(2)

5.42.4.2 pdnSyslogIPAddr

This value is the IP address of a syslog server. If the ip address is known then this can be used for the remote syslog daemon.

5.42.4.3 pdnSyslogLevel

This value corresponds to a minimum severity level of events that will be logged. The following values are defined for this object:

- emerg(1)
- err(2)
- norm(3)
- info(4)

The default value is norm(3).

NOTE:

This object has been deprecated. It is still valid on older products that use this mib,

such as DSL Router endpoints.

All new products should use pdnSyslogSeverityThreshold (Section 5.42.4.5, Page 493) which uses the severity values in the RFC 3164.

5.42.4.4 pdnSyslogPort

This value corresponds to a UDP port number to which system events are sent.

The default is 514.

5.42.4.5 pdnSyslogSeverityThreshold

This value corresponds to a minimum severity level that a syslog messages can be set. All syslog message with an enum value greater than the threshold will not be sent. The values are obtained from RFC 3164, “The BSD Syslog Protocol.” The following are valid values:

- emerg(0) - Emergency: system is unusable
- alert(1) - Alert: action must be taken immediately
- critical(2) - Critical: critical conditions
- error(3) - Error: error conditions
- warning(4) - Warning: warning conditions
- notice(5) - Notice: normal but significant condition
- info(6) - Informational: informational messages
- debug(7) - Debug: debug-level messages

5.42.4.6 pdnSyslogRemoteDaemon

This value allows the user to disable or enable syslog messages going to a remote daemon. The following values are defined for this object:

- disable(1)
- enable(2)

5.42.4.7 pdnSyslogTable

This table supports retrieval of the syslog messages.

Index:

- pdnSyslogNumber

OIDs:

- pdnSyslogTable - { pdnSyslog 7 } 1.3.6.1.4.1.1795.2.24.2.31.1.7
- pdnSyslogEntry - { pdnSyslogTable 1 } 1.3.6.1.4.1.1795.2.24.2.31.1.7.1

Table 342. pdnSyslogTable

Object	OID	Syntax	Access	Status	Supported
pdnSyslogNumber	{ pdnSyslogEntry 1 }	Integer32	not-accessible	current	Y
pdnSyslogMessage	{ pdnSyslogEntry 2 }	OCTET STRING(SIZE(1024))	read-only	current	Y

5.42.4.7.1 pdnSyslogNumber

This is a sequentially increasing index of syslog messages starting at 1. When the table reaches its maximum size as defined by pdnSyslogMaxTableSize (Section 5.42.4.9, Page 494) then the oldest message is dropped without renumbering any messages that remain. The index is reset to 1 on a device reset or when the table is cleared using pdnSyslogClearTable (Section 5.42.4.10, Page 494)pdnSyslogClearTable.

5.42.4.7.2 pdnSyslogMessage

The text of the syslog message.

5.42.4.8 pdnSyslogNumOfMsgInTable

This value lets the users know how many syslog messages are currently in the syslog table.

5.42.4.9 pdnSyslogMaxTableSize

The maximum number of syslog message the syslog table can hold.

5.42.4.10 pdnSyslogClearTable

Clears all the entries in the syslog message table. The following values are defined for this object:

- noOp(1)
- clear(2)

5.42.4.11 pdnSyslogMsgToConsole

This allows messages to be sent to the device's console port. Note this is intended for debug use only! It is not recommended to be used to monitor syslog message on a regular bases. If one needs to constantly monitor the syslog messages, then they should use the remote syslog daemon, or other user interfaces. The following values are defined for this object:

- disable(1)
- enable(2)

5.42.4.12 pdnSyslogRateLimiting

This allows one to enable or disable rate limiting. When rate limiting is enabled, it prevents the exact same message from being resent within a fixed amount of time. The following values are defined for this object:

- disable(1)
- enable(2)

5.43 PDN-DEVICE-TIME-MIB — pdn_time.mib

5.43.1 Introduction

This MIB consists for objects for date and time functions.

The version of the MIB used for this document had a comment stating it was last updated on June 3, 2003.

5.43.2 Textual Conventions

The following “special” textual conventions are made or referenced in this MIB:

- DateAndTime - A date-time specification. The display hint is “2d-1d-1d,1d:1d:1d.1d,1a1d:1d.”

Field	Octets	Contends	Range
1	1-2	year[a]	0…65536
2	3	month	1…12
3	4	day	1…31
4	5	hour	0…23
5	6	minutes	0…59
6	7	seconds	0…60[b]
7	8	deci-seconds	0…9
8	9	direction from UTC	“+” or “-”
9	10	hours from UTC[c]	0…13
10	11	minutes from UTC	0…59

a. The value if year is in network-byte order

b. Use 60 for leap-second

c. Daylight saving time in New Zealand is +13

  For example, Tuesday May 26, 1992 at 1:30:15 PM EDT would be displayed as: 1992-5-26,13:30:15.0,-4:0.
  Note that if only local time is known, then timezone information (fields 8-10) is not present.

  The syntax for this object is:

  — OCTET STRING (SIZE (8 | 11))

- IpAddress - This application-wide type represents a 32-bit internet address. The syntax for this object is:
  — OCTET STRING of length 4, in network byte-order

- NTPMode - This object describes the mode in which NTP/SNTP will operate.

  The syntax for this object is:

  — INTEGER
    - unicast(1)
    - broadcast(2)
    - multicast(3)

OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdn-time - { pdn-common 20 } 1.3.6.1.4.1.1795.2.24.2.20
- devTimeMibObjects - { pdn-time 1 } 1.3.6.1.4.1.1795.2.24.2.20.1
- devTimeMibTraps - { pdn-time 2 } 1.3.6.1.4.1.1795.2.24.2.20.2
- devTimeAndDate - { devTimeMibObjects 1 } 1.3.6.1.4.1.1795.2.24.2.20.1.1
- devNTP - { devTimeMibObjects 2 } 1.3.6.1.4.1.1795.2.24.2.20.1.2

5.43.3 MIB Objects

Table 343. PDN-DEVICE-TIME-MIB Objects

Object	OID	Description	Supported
devTimeAndDate	{ devTimeMibObjects 1 }	Date and Time Group	N
devNTP	{ devTimeMibObjects 2 }	Simple Network Time Protocol Group	Y

5.43.3.1 devTimeAndDate

The Date and Time Group.

OIDs:

- devTimeAndDate - { devTimeMibObjects 1 } 1.3.6.1.4.1.1795.2.24.2.20.1.1

Table 344. devTimeAndDate

Object	OID	Syntax	Access	Status	Supported
devDateAndTime	{ devTimeAndDate 1 }	DateAndTime	read-write	mandatory	N

5.43.3.1.1 devDateAndTime

This object displays the date and time on the entity. It allows an NMS to set the date and time in the appropriate timezone.

5.43.3.2 devNTp

The Simple Network Time Protocol Group.

OIDs:

- devNTP - { devTimeMibObjects 2 } 1.3.6.1.4.1.1795.2.24.2.20.1.2

Table 345. devNTp

Object	OID	Syntax	Access	Status	Supported
devNTPServerIP	{ devNTP 1 }	IpAddress	read-write	mandatory	Y
devNTPMode	{ devNTP 2 }	NTPMode	read-only[a]	mandatory	Y
devNTPSynchronised	{ devNTP 3 }	INTEGER(1…24)	read-write	mandatory	Y
devNTPEnable	{ devNTP 4 }	INTEGER enabled(1) disabled(2)	read-write	mandatory	Y
devNTPOffsetFromUTC	{ devNTP 5 }	INTEGER	read-write	mandatory	Y

a. Product imposed limit.

5.43.3.2.1 devNTPServerIP

The IP address of the NTP/SNTP server.

Valid Object Value(s)

The default value for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM is 192.5.41.40.

5.43.3.2.2 devNTPMode

This object allows an NMS to configure the mode in which NTP/SNTP will operate. In unicast(1) mode, NTP/SNTP polls the server specified via the devNTPServerIP object. In broadcast(2) mode, NTP/SNTP polls by broadcasting a request so that any NTP/SNTP server could respond. Alternatively, in broadcast(2) mode, the client only listens for messages broadcasted by any server. In multicast(3) mode, the client listens for messages from any server sent to the multicast group defined for NTP/SNTP (224.0.1.1).

Default value is broadcast(2).

Valid Object Value(s)

The only valid value for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM is unicast(1) mode. Therefore write operations to this object will not be supported.

5.43.3.2.3 devNTPSynchronised

This object allows an NMS to configure the poll interval (hours) for the NTP/SNTP client to use to send requests to an NTP/SNTP server to synchronize the local clock for the device.

Default value is 1 hour.

Valid Object Value(s)

The default value for the BitStorm 2600 and GranDSLAM 4200 IP DSLAM is 24 hours.

5.43.3.2.4 devNTPEnable

The following values are defined for this object:

Indicates whether or not the local clock is synchronized via NTP/SNTP. The following values are defined for this object:

- enabled(1)
- disabled(2)

When set to enabled(1), NTP/SNTP is operational and will attempt to synchronize the local clock based on the message(s) received from an NTP/SNTP server.

Default value is enabled(1).

5.43.3.2.5 devNTPOffsetFromUTC

Offset (minutes) from the coordinated universal time (UTC), also known as Greenwich mean time (GMT), to local time represented as a signed integer. The sign of the integer denotes the direction of local time from UTC: offsets for local times west of the zero meridian to the international date line are negative.

Default value is 0 (that is, equal to UTC).

5.44 PDN-UPLINK-TAGGING-MIB — PDN-UPLINK-TAGGING-MIB.mib

5.44.1 Introduction

This MIB contains objects that are used to configure Uplink Tagging (ULT). Uplink Tagging is a term used to describe a feature that simplifies the setup and administration of networks where a service provider wants to use a unique VLAN per subscriber port. Ingress frames are tagged with a VLAN and these tagged frames are transmitted on the uplink port towards the WAN.

In cases where the hardware implementation permits, multiple units can be interconnected together to form a "Uplink Tagging Domain (ULT Domain)." A ULT domain is defined as the set of interconnected Paradyne DSLAMs that share a common block of VLAN IDs. The maximum number of Paradyne DSLAMs that can be interconnected is implementation dependent. Generally, all DSLAMs in a ULT Domain are configured with the same block of VLAN IDs. Each chassis/unit is assigned a unique ULT Index within the ULT Domain.

There are two parts of configuring Uplink Tagging:

1. Uplink Base VLAN Tag-This object specifies the beginning VLAN ID for a particular common block of VLAN IDs. This object is defined as an enumeration whose values will depend on the number of port in a chassis/unit.

2. Uplink Tag Index-This object specifies the index within some block of VLAN IDs. Generally, this index can thought of a chassis/unit number as can be seen with the examples below.

When configured in Uplink Tagging Mode, the ipNetToMediaForwardingMode MIB object, Section 5.22.4.4.1, Page 335, is set to ult(4). In this mode, the user configures VLANs indirectly by using the following MIB objects defined in this MIB:

- pdnUltIndex - This new object is basically an Unsigned32 that excludes 0. It idea is that different implementations will support different maximum values of the index. As such, the syntax for this object will cover any implementation and actual implementation specific maximum values should be documented in something like the implementation's SNMP Op Spec.
- pdnGenUltBaseVlanTag - This object allows a any base VLAN Tag to be defined.
- pdn48UltBaseVlanTag - This object defines a set of enumerations for base VLAN Tags for chassis/units that contain 48 ports.
- pdn24UltBaseVlanTag - This object defines a set of enumerations for base VLAN Tags for chassis/units that contain 24 ports.

The version of the MIB used for this document had a LAST-UPDATED of 200303120000Z which decodes to March 12, 2003.

5.44.2 Textual Conventions

No "special" textual conventions are made or referenced in this MIB.

5.44.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1
- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2

- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnUplinkTagging - { pdn-common 37 } 1.3.6.1.4.1.1795.2.24.2.37
- pdnUplinkTaggingObjects - { pdnUplinkTagging 1 } 1.3.6.1.4.1.1795.2.24.2.37.1
- pdnUplinkTaggingConformance - { pdnUplinkTagging 2 } 1.3.6.1.4.1.1795.2.24.2.37.2
- pdnUplinkTaggingObjectsR2 - { pdnUplinkTagging 3 } 1.3.6.1.4.1.1795.2.24.2.37.3

5.44.4 MIB Objects

Table 346. PDN-STACKABLE-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnUltIndex	{ pdnUplinkTaggingObjectsR2 1 }	Unsigned32(1…4294967295)	read-write	Current	Y
pdnGenUltBaseTag	{ pdnUplinkTaggingObjectsR2 2 }	VlanId	read-write	Current	Y
pdn48UltBaseVlanTag	{ pdnUplinkTaggingObjectsR2 3 }	INTEGER ultBase16(1) ultBase512(2) ultBase1024(3) ultBase1536(4) ultBase2048(5) ultBase2560(6) ultBase3072(7) ultBase3584(8)	read-write	Current	Y
pdn24UltBaseVlanTag	{ pdnUplinkTaggingObjectsR2 3 }	INTEGER ultBase16(1) uldBase256(2) ultBase512(3) ultBase768(4) ultBase1024(5) ultBase1280(6) ultBase1536(7) ultBase1792(8) ultBase2048(9) ultBase2304(10) ultBase2560(11) ultBase2816(12) ultBase3072(13) ultBase3328(14) ultBase3584(15) ultBase3840(16)	read-write	Current	Y
ultBaseVlanTag	{ pdnUplinkTaggingObjects 1 }	INTEGER ultBase16(1) ultBase512(2) ultBase1024(3) ultBase1536(4) ultBase2048(5) ultBase2560(6) ultBase3072(7) ultBase3584(8)	read-write	deprecated	N
ultIndex	{ pdnUplinkTaggingObjects 2 }	INTEGER(1…5)	read-write	deprecated	N

5.44.4.1 pdnUltIndex

This object represents VLAN tag index which is an index into a block of VLAN Tags the unit will use. Generally, this can be also thought of as the chassis/unit number in the case where multiple units are interconnected and form a ULT Domain described above.

DEFVAL- 1

Valid Object Value(s)

The upper limit for this object is:

- 5 for 48-port units
- 10 for 24-port units

5.44.4.2 pdnGenUltBaseVlanTag

This object can be used to allow any Uplink Tagging Base Index to be entered when they don't like the "canned" list defined in the objects below.

DEFVAL-16

5.44.4.3 pdn48UltBaseVlanTag

This object represents Uplink Tagging base index which is the starting VLAN ID for a particular common block of VLAN IDs for chassis/units that contain 48 DSL subscriber ports. The following are values are defined for this object:

- ultBase16(1) - 16 (0x0010)
- ultBase512(2) - 512 (0x0200)
- ultBase1024(3) - 1024 (0x0400)
- ultBase1536(4) - 1536 (0x0600)
- ultBase2048(5) - 2048 (0x0800)
- ultBase2560(6) - 2560 (0x0A00)
- ultBase3072(7) - 3072 (0x0C00)
- ultBase3584(8) - 3584 (0x0E00)

DEFVAL-ultBase16

Table 347. ULT Base VLAN Tags to Unit Relationship for Units Containing 48-Ports

ULT Base VLAN Tag		ULT Tags (decimal)				
(decimal)	(hex)	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
16	0x0010	16 - 63	64 - 111	112 - 159	160 - 207	208 - 255
512	0x0200	512 - 559	560 - 607	608 - 655	656 - 703	704 - 751
1024	0x0400	1024 - 1071	1072 - 1119	1120 - 1167	1168 - 1215	1216 - 1263
1536	0x0600	1536 - 1583	1584 - 1631	1632 - 1679	1680 - 1727	1728 - 1775
2048	0x0800	2048 - 2095	2096 - 2143	2144 - 2191	2192 - 2239	2240 - 2287
2560	0x0A00	2560 - 2607	2608 - 2655	2656 - 2703	2704 - 2751	2752 - 2799
3072	0x0C00	3072 - 3119	3120 - 3167	3168 - 3215	3216 - 3263	3264 - 3311

Table 347. ULT Base VLAN Tags to Unit Relationship for Units Containing 48-Ports

ULT Base VLAN Tag		ULT Tags (decimal)				
(decimal)	(hex)	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
3584	0x0E00	3584 - 3631	3632 - 3679	3680 - 3727	3728 - 3775	3776 - 3823

5.44.4.4 pdn24UltBaseVlanTag

This object represents Uplink Tagging base index which is the starting VLAN ID for a particular common block of VLAN IDs for chassis/units that contain 24 DSL subscriber ports.

The following are values are defined for this object:

- ultBase16(1) - 16 (0x0010)
- ultBase256(2) - 126 (0x0100)
- ultBase512(3) - 512 (0x0200)
- ultBase768(4) - 768 (0x0300)
- ultBase1024(5) - 1024 (0x0400)
- ultBase1280(6) - 1280 (0x0500)
- ultBase1536(7) - 1536 (0x0600)
- ultBase1792(8) - 1792 (0x0700)
- ultBase2048(9) - 2048 (0x0800)
- ultBase2304(10) - 2304 (0x0900)
- ultBase2560(11) - 2560 (0x0A00)
- ultBase2816(12) - 2816 (0x0B00)
- ultBase3072(13) - 3072 (0x0C00)
- ultBase3328(14) - 3328 (0x0D00)
- ultBase3584(15) - 3584 (0x0E00)
- ultBase3840(16) - 2840 (0x0F00)

Table 348. ULT Base VLAN Tags to Unit(1 to 5 of 10) Relationship for Units Containing 24-Ports

ULT Base VLAN Tag		ULT Tags (decimal)				
(decimal)	(hex)	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
16	0x0010	16 - 39	40 - 63	64 - 87	88 - 111	112 - 135
256	0x0100	256 - 279	280 - 303	304 - 327	328 - 351	352 - 375
512	0x0200	512 - 535	536 - 559	560 - 583	584 - 607	608 - 631
768	0x0300	768 - 791	792 - 815	816 - 839	840 - 863	864 - 887
1024	0x0400	1024 - 1047	1048 - 1071	1072 - 1095	1096 - 1119	1120 - 1143
1280	0x0500	1280 - 1303	1304 - 1327	1328 - 1351	1352 - 1375	1375 - 1399
1536	0x0600	1536 - 1559	1560 - 1583	1584 - 1607	1608 - 1631	1632 - 1655

Table 348. ULT Base VLAN Tags to Unit(1 to 5 of 10) Relationship for Units Containing 24-Ports

ULT Base VLAN Tag		ULT Tags (decimal)				
(decimal)	**(hex)**	**Unit 1**	**Unit 2**	**Unit 3**	**Unit 4**	**Unit 5**
1792	0x0700	1792 - 1815	1816 - 1839	1840 - 1863	1864 - 1887	1888 - 1911
2048	0x0800	2048 - 2071	2072 - 2095	2096 - 2119	2120 - 2143	2144 - 2167
2304	0x0900	2304 - 2327	2328 - 2351	2352 - 2375	2376 - 2399	2400 - 2423
2560	0x0A00	2560 - 2583	2584 - 2607	2608 - 2631	2632 - 2655	2656 - 2679
2816	0x0B00	2816 - 2839	2840 - 2863	2864 - 2887	2888 - 2911	2912 - 2935
3072	0x0C00	3072 - 3095	3096 - 3119	3120 - 3143	3144 - 3167	3168 - 3191
3328	0x0D00	3328 - 3351	3352 - 3375	3376 - 3399	3400 - 3423	3424 - 3447
3584	0x0E00	3584 - 3607	3608 - 3631	3632 - 3655	3656 - 3679	3680 - 3703
3840	0x0F00	3840 - 3863	3864 - 3887	3888 - 3911	3912 - 3935	3936 - 3959

Table 349. ULT Base VLAN Tags to Unit(6 to 10 of 10) Relationship for Units Containing 24-Ports

ULT Base VLAN Tag		ULT Tags (decimal)				
(decimal)	**(hex)**	**Unit 6**	**Unit 7**	**Unit 8**	**Unit 9**	**Unit 10**
16	0x0010	136 - 159	160 - 183	184 - 207	208 - 231	232 - 255
256	0x0100	376 - 399	400 - 423	424 - 447	448 - 471	472 - 495
512	0x0200	632 - 655	656 - 679	680 - 703	704 - 727	728 - 751
768	0x0300	888 - 911	912 - 935	936 - 959	960 - 983	984 - 1007
1024	0x0400	1144 - 1167	1168 - 1191	1192 - 1215	1216 - 1239	1240 - 1263
1280	0x0500	1400 - 1423	1424 - 1447	1448 - 1471	1472 - 1495	1496 - 1519
1536	0x0600	1656 - 1679	1680 - 1703	1704 - 1727	1728 - 1751	1752 - 1775
1792	0x0700	1912 - 1935	1936 - 1959	1960 - 1983	1984 - 2007	2008 - 2031
2048	0x0800	2168 - 2191	2192 - 2215	2216 - 2239	2240 - 2263	2264 - 2287
2304	0x0900	2424 - 2447	2448 - 2471	2472 - 2495	2496 - 2519	2520 - 2543
2560	0x0A00	2680 - 2703	2704 - 2727	2728 - 2751	2752 - 2775	2776 - 2799
2816	0x0B00	2936 - 2959	2960 - 2983	2984 - 3007	3008 - 3033	3034 - 3057
3072	0x0C00	3192 - 3215	3216 - 3239	3240 - 3263	3264 - 3287	3288 - 3311
3328	0x0D00	3448 - 3471	3472 - 3495	3496 - 3519	3520 - 3543	3544 - 3567
3584	0x0E00	3704 - 3727	3728 - 3751	3752 - 3775	3776 - 3799	3800 - 3823
3840	0x0F00	3960 - 3983	3984 - 4007	4008 - 4031	4032 - 4055	4056 - 4079

5.44.4.5 ultBaseVlanTag

This object has been deprecated.

This object represents Uplink Tagging base index. The following are values are defined for this object:

- ultBase16(1) - 16 (0x0010)
- ultBase512(2) - 512 (0x0200)
- ultBase1024(3) - 1024 (0x0400)
- ultBase1536(4) - 1536 (0x0600)
- ultBase2048(5) - 2048 (0x0800)
- ultBase2560(6) - 2560 (0x0A00)
- ultBase3072(7) - 3072 (0x0C00)
- ultBase3584(8) - 3584 (0x0E00)

The default value for this object is ultBase16(1).

5.44.4.6 ultIndex

This object has been deprecated.

This object represents VLAN tag index which represents an index into a block of VLAN Tags the unit will use.

The default value for this object is “1.”

5.45 PDN-VLAN-MIB — PDN-VLAN-MIB.mib

5.45.1 Introduction

This MIB module contains objects pertaining to VLANs.

The revision of the MIB that this SNMP Operational Specification aligns with is that which has a LAST-UPDATED of “200304110000Z,” or March 11, 2003.

5.45.2 Textual Conventions

No “special” textual conventions are made or referenced in this MIB.

5.45.3 OIDs

- iso - { 1 } 1
- org - { iso 3 } 1.3
- dod - { org 6 } 1.3.6
- internet - { dod 1 } 1.3.6.1

- mgmt - { internet 2 } 1.3.6.1.2
- mib-2 - { mgmt 1 } 1.3.6.1.2.1
- ip - -{ mib-2 4 } 1.3.6.1.2.1.4

- private - { internet 4 } 1.3.6.1.4
- enterprises - { private 1 } 1.3.6.1.4.1
- pdyn - { enterprises 1795 } 1.3.6.1.4.1.1795
- pdn-mgmt - { pdyn 2 } 1.3.6.1.4.1.1795.2
- paradyne - { pdn-mgmt 24 } 1.3.6.1.4.1.1795.2.24
- pdn-common - { paradyne 2 } 1.3.6.1.4.1.1795.2.24.2

- pdnVlanMIB - { pdn-common 46 } 1.3.6.1.4.1.1795.2.24.2.46
- pdnVlanNotifications - { pdnVlanMIB 0 } 1.3.6.1.4.1.1795.2.24.2.46.1
- pdnVlanObjects - { pdnVlanMIB 1 } 1.3.6.1.4.1.1795.2.24.2.46.1

5.45.4 MIB Objects

Table 350. PDN-VLAN-MIB Objects

Object	OID	Syntax	Access	Status	Supported
pdnVlanReservedBlockStart	{ pdnVlanObjects 1 }	VlanIndex	read-write	current	Y
pdnVlanInbandMgmtVlanId	{ pdnVlanObjects 2 }	VlanIndex	read-write	current	N

5.45.4.1 pdnVlanReservedBlockStart

This object defines the starting VLAN for a sequential block of VLANS that are to be reserved for internal use. The actual size of the block reserved is not specified as it could be implementation specific. The size of the actual block being reserved should be clearly specified in the SNMP Operational Specification for the particular implementation.

Valid Object Value(s)

The size of this block is up to 16 sequential VLAN IDs. The only case where the size of this block is less than 16 would be if it began at the upper VLAN ID where there were not 16 sequential VLAN ID available. For example, if the block started at 4080, than there would only be 14 VLAN IDs reserved (4080 through 4094 inclusive).

5.45.4.2 pdnVlanInbandMgmtVlanId

The VLAN ID assigned to the In-Band Management Channel.

5.45.5 Compliance Statements

5.45.5.1 Compliance

5.45.5.1.1 pdnVlanMIBCompliance

The compliance statement for the pdnVlan entities which implement the pdnVlanMIB.

Mandatory Groups:

None specified.

Optional Groups:

- pdnVlanReservedBlockGroup - This group is mandatory only in those entities which implement reserving a single sequential block of VLANs for internal purposes.
- pdnVlanInbandMgmtVlanGroup - This group is mandatory only in those entities which require the VLAN for the In-Band Management channel to be configured and identified.

5.45.5.2 Conformance

Table 351. PDN-VLAN-MIB Conformance

Group	Description	Objects	Supported
pdnVlanReservedBlockGroup	Objects grouped for reserving a block of sequential VLANs.	pdnVlanReservedBlockStart	Y
pdnVlanInbandMgmtVlanGroup	Objects grouped relating to the In-Band Management VLAN.	pdnVlanInbandMgmtVlanId	Y

Index

L

M