Hotwire 8620 and 8820

GranDSLAM

SNMP Reference

Document No. 8000-A2-GB30-00

June 2003

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

Contents

About This Guide

Document Purpose and Intended Audience

This reference manual contains the network management specification for the Hotwire GranDSLAM 8620 and 8820 Release 3.0 and above. It is designed for network managers who use SNMP in the course of installing, configuring, monitoring, and troubleshooting GranDSLAM chassis and associated cards.

Document Summary

Section	Description
Chapter 1, Network Management Specification	Provides detailed information about the network management interface of the GranDSLAM 8620 and 8820 chassis.
Index	Lists key terms, concepts, and sections in alphabetical order.

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* → *Hotwire DSL Systems*.

Document Number	Document Title
7890-A2-GB22	*GrandVIEW EMS User's Guide*
8400-A2-GB20	*Hotwire SCP User's Guide*
8620-A2-GN20	*Hotwire 8620 GranDSLAM Installation Guide*
8820-A2-GN20	*Hotwire 8820 GranDSLAM Installation Guide*

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

Network Management Specification

1

1. GranDSLAM 3.0 Network Management Interface

This document describes the SNMP support for the GranDSLAM 3.0 product. The product includes the following models:

Table 1-1. Model Numbers for the GranDSLAM 3.0 Product

Type	Model Description	Release
SHDSL	8985 ATM Card, 24 SHDSL Ports	R3.0
ReachDSL-V3	8955 ATM Card, 24 ReachDSL-V3 Ports	R3.0
ADSL	8965 ATM Card, 24 ADSL Ports	R3.0
SCP	8412-A1-000, SCP with OC3/STM1 MM ATM uplink	R3.0
SCP	8413-A1-000, SCP with OC3/STM1 SM int ATM uplink	R3.0
SCP	8414-A1-000, SCP with OC3/STM1 SM long ATM uplink	R3.0
SCP	8416-A1-000, SCP with 8 T1 IMA Uplink	R3.0
SCP	8417-A1-000, SCP with 8 E1 IMA Uplink 75 ohm	R3.0
SCP	8418-A1-000, SCP with 8 E1 IMA Uplink 120 ohm	R3.0

These GranDSLAM 3.0 model units are configurable from SNMP or through the TL1 Command Line Interface using the following MIBS.

Table 1-2. Supported RFCs and MIBs (1 of 5)

RFC	Name	Description
1213	MIB-II	Management Information Base for Network Management of TCP/IP-based internets [system_t]
2737	Entity MIB (Version 2)	Definitions of Managed Objects for Entity MIB [entPhysicalEntry_t], [entLogicalEntry_t], [entAliasMappingEntry_t], [entPhysicalContainsEntry_t],[entityGeneral_t]

Table 1-2. Supported RFCs and MIBs (2 of 5)

RFC	Name	Description
2495	Definitions of Managed Objects for the DS1, E1, DS2 and E2 Interface Types	Definitions of Managed Objects for the DS1, E1, DS2 and E2 Interface Types [dsx1ConfigEntry_t],[dsx1CurrentEntry_t], [dsx1IntervalEntry_t], [dsx1TotalEntry_t],[dsx1ChanMappingEntry_t], [dsx1FarEndCurrentEntry_t],[dsx1FarEndIntervalEntry_t], [dsx1FarEndTotalEntry_t]
2514	Definitions of Textual Conventions for ATM Management	Definitions of Textual Conventions and OBJECT-IDENTITIES for ATM Management
2515	Definitions of Managed Objects for ATM Management [AToM]	ATM MIB (Uses Definitions from 2514) [atmMIBObjects_t], [atmInterfaceConfEntry_t],[atmInterfaceTCEntry_t], [atmTrafficDescrParamEntry_t],[atmVclEntry_t], [atmVcCrossConnectEntry_t]]
2863	Evolution of MIB-II Interfaces	Evolution of the Interfaces Group of MIB-II [interfaces_t], [ifEntry_t],[ifStackEntry_t]
2864	Inverted Stack Extension MIB	Definitions of Managed Objects for the Inverted Stack Table Extension to the Interfaces Group MIB: [ifInvStackEntry_t]
2662	Definitions of Managed Objects for the ADSL Lines	Definitions of Managed Objects for the ADSL Lines. Note: Used for the DMT and Rv3 product line [adslLineEntry_t],[adslAtucPhysEntry_t],[adslAturPhysEntry_t], [adslAtucChanEntry_t],adslAturChanEntry_t], [adslAtucPerfDataEntry_t],[adslAturPerfDataEntry_t], [adslAtucIntervalEntry_t],[adslAturIntervalEntry_t], [adslAtucChanIntervalEntry_t],adslAturChanIntervalEntry_t], [adslLineConfProfileEntry_t],[adslLineAlarmConfProfileEntry_t]
2662 supplement	draft-ieft-adslmib-adslext-xx.txt where: xx is Latest Version (http://www.ietf.cnri.reston.va.us/internet-drafts/)	Definitions of Extension Managed Objects for ADSL Lines : [adslLineExtEntry_t],[adslAtucPerfDataExtEntry_t], [adslAturPerfDataExtEntry_t],adslConfProfileExtEntry_t], [adslAlarmConfProfileExtEntry_t]
2096	IP Forwarding Table	Definitions of Managed objects for the Ip Forwading table for classless inter-domain routing (CIDR). [ipCidrRouteNumber], [ipCidrRouteTable]

Table 1-2. Supported RFCs and MIBs (3 of 5)

RFC	Name	Description
N/A	IMA MIB AF-PHY-0086.001 (http://www.atmforum.com/atmforum/specs/approved.html)	The MIB module for managing ATM Forum Inverse Multiplexing for ATM (IMA) interfaces. [imaGroupNumber],[imaGroupEntry],[imaGroupMappingEntry], [ImaLinkEntry],[ImaAlarmType],[imaGroupCurrentEntry],[ImaGroup IntervalEntry],[ImaGroupTotalEntry],[imaLinkCurrentEntry],[imaLinkTotalEntry],[ImaLinkIntervalEntry]
N/A	SNMP M4 Network Element View MIB [atmfM4MIB] AF-NM-0095.001 (http://www.atmforum.com/atmforum/specs/approved.html)	ATM Forum SNMP M4 Network Element View [atmfM4MIBObjects_t],[atmfM4TcAdapterEntry_t], [atmfM4ATMLayerEntry_t],[atmfM4VpNextVpiEntry_t], [atmfM4VcNextVciEntry_t],[atmfM4CellProtoCurrEntry_t], [atmfM4CellProtoHistEntry_t],[atmfM4CellProtoErrorEntry_t], [atmfM4TcProtoCurrEntry_t],[atmfM4TcProtoHistEntry_t], [atmfM4VcUpcNpcCurrEntry_t], [atmfM4VcUpcNpcHistEntry_t],[atmfM4VcTestEntry_t]
2668	Definitions of Managed objects for IEEE 802.3 Medium Attachment Units(MAU).	ifMauEntry_t
2571	Architecture for SNMP Frameworks	snmpEngineID, snmpEngineBoots, snmpEngineTime, snmpEngineMaxMessageSize
2572	Message Processing and Dispatching for the Simple Network Management Protocol	snmpUnknownSecurityModels, snmpInvalidMsgs, snmpUnknownPDUHandlers
2573	SNMP Applications (SNMPv3)	snmpTargetAddrEntry_t, snmpTargetParamsTable_t, snmpUnavailableContexts, snmpUnknownContexts, snmpNotifyEntry_t
2573	User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)	usmStats, usmUserEntry_t
2575	View-based Access Control Model (VACM) for the version 3 of the Simple Network Management Protocol (SNMPv3)	vacmContextEntry_t, vacmSecurityToGroupEntry_t, vacmAccessEntry_t, vacmViewTreeFamilyEntry_t,
2576	Coexistence between Version1, Version 2, and Version 3 of the Internet-standard Network Management Framework	snmpCommunityEntry_t, snmpTargetAddrExtEntry_t
2558	Definitions of Managed Objects for the SONET/SDH Interface Type	sonetMediumEntry_t, sonetLineCurrentEntry_t, sonetLineIntervalEntry_t, sonetSectionCurrentEntry_t, sonetSectionIntervalEntry_t,sonetFarEndLineCurrentEntry_t, sonetFarEndLineIntervalEntry_t, sonetPathCurrentEntry_t, sonetPathIntervalEntry, sonetPathFarEndCurrentEntry_t, sonetPathFarEndIntervalEntry_t
N/A	Definitions of Managed Objects for SONET Linear APS architectures. draft-ieft-atommib-sonetaps-mib-xx.txt	apsConfigEntry_t, apsStatusEntry_t, apsMapEntry_t, apsChanConfigEntry_t, apsCommandEntry_t, apsChanStatusEntry_t, apsNotificationEnable

Table 1-2. Supported RFCs and MIBs (4 of 5)

RFC	Name	Description
3276	Definitions of Managed Objects for High Bit-Rate DSL -2nd generation (HDSL2) and Single-Pair High-Speed Digital Subscriber Line (SHDSL) Lines	hdsl2ShdslSpanConfEntry_t, hdsl2ShdslSpanStatusEntry_t, hdsl2ShdslInventoryEntry_t, hdsl2ShdslEndpointConfEntry_t, hdsl2ShdslEndpointCurrEntry_t, hdsl2Shdsl15MinIntervalEntry_t, hdsl2Shdsl1DayIntervalEntry_t, hdsl2ShdslEndpointMaintEntry_t, hdsl2ShdslUnitMainEntry_t, hdsl2ShdslSpanConfProfileEntry_t, hdsl2ShdslEndpointAlarmConfProfileEntry_t
3433	Definitions of Managed Objects for Entity Sensor Management Information Base	entPhySensorEntry_t
enterprise	pdn_SonetExt.mib	Paradyne Enterprise Sonet Extension MIB devSonetConfigEntry_t
enterprise	PDN-ENTITY-REDUNDANCY-MIB.mib	Paradyne Enterprise Entity Redundancy MIB pdnEntityRedundancySelection, pdnRedunCmdEntry, pdnRedunStatusEntry
enterprise	PDN-LINK-FAULT-MGMT-MIB.mib	Paradyne Enterprise Link Fault Management MIB pdnLinkFaultMgmtApsSelection, pdnLinkFaultMgmtSwitchoverSelection, pdnLinkFailureConfigEntry,
enterprise	PDN-LINK-LOAD-SHARING-MIB.mib	Paradyne Enterprise Load Sharing MIB pdnLinkLoadSharingSelection pdnLLSConfigEntry, pdnLLSMapEntry
enterprise	mpe_Control.mib	Paradyne Enterprise Device Control MIB includes the Paradyne File Transfer Group. [mpeDevFileXferConfigEntry_t], [mpeDevHwControlEntry_t], [mpeDevFirmwareControlEntry_t], [mpeDevHwControl_t]
enterprise	pdn_tc.mib	Paradyne Enterprise Textual Convention MIB
enterprise	mpe_HealthAndStatus.mib	Paradyne Enterprise Health and Status MIB mpeDevHealthAndStatusEntry_t
enterprise	pdn_ATMExt.mib pdn_ATMM4Ext.mib pdn_ATMStats.mib	Paradyne Enterprise ATM mibs [pdnATMIfConfExtEntry_t], [pdnATMTrafficDescrParamExtEntry_t], [pdnATMfM4TcProtoCurrExtEntry_t], [pdnATMfM4TcProtoHistExtEntry_t], [pdnATMVclHistEntry_t], [pdnATMVclCurrEntry_t],
enterprise	PDN-MGMT-IP-MIB.mib	Paradyne Management of IP over ATM interfaces pdnMgmtIpMib
enterprise	mpe_Config.mib	mpeDevConfigClockSrcEntry_t

Table 1-2. Supported RFCs and MIBs (5 of 5)

RFC	Name	Description
enterprise	pdn_Config.mib	devConfigTestTimer
enterprise	pdn_Security.mib	[devSecurityMgrValidation], [devSecurityMgrMaxNumber], [devSecurityMgrCurrentNumber], [securityMgrTable]
enterprise	pdn_time.mib	[devTimeAndDate], [devNTP]
enterprise	pdn_syslog.mib	[pdnSyslog], [pdnSyslogEntry]
enterprise	pdn_dslam.mib	[sysDevUserAccountEntry]
enterprise	pdn_spectrummgmt.mib	Paradyne spectrum management specific mib. newSpectrumMgmtGeneralConfigEntry, newSpectrumMgmtConfEntry, newSpectrumMgmtLineInfoEntry
enterprise	pdn_reachDSL.mib	Paradyne ReachDSL specific mib. reachDSLSpectrumMgmtSelection reachDSLSpectrumMgmtConfEntry reachDSLSpectrumMgmtLineInfoEntry reachDSLLineEntry reachDSLSpectrumMgmtLoopMeasurementMethod reachDSLSpectrumMgmtEWLUnits reachDSLSpectrumMgmtMod
enterprise	PDN-DS1EXT-MIB	Paradyne Enterprise supplement for RFC 2495 pdnDs1ExtConfEntry_t
enterprise	mpe_dslam	mpeEntPhysicalExtEntry_t

Table 1-3. Other Supported RFCs Used as Guidelines for Implementation

RFC	Name/Description
768	User Datagram Protocol
791	Internet Protocol
792, 950	Internet Control Message Protocol; Internet Standard Subnetting Procedure
793	Transmission Control Protocol
826	Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware
854	Telnet Protocol Specification
894	A Standard for the Transmission of IP Datagrams over Ethernet Networks
959	File Transfer Protocol
1042	Transmission of IP datagrams over IEEE 802.3 networks
1122	Requirements for Internet Hosts Communication Layers
1123	Requirements for Internet Hosts Application and Support
1157	Simple Network Management Protocol (SNMP) SNMP version 1
1483, 2684	Multiprotocol Encapsulation over ATM Adaptation Layer 5
1659	Definitions of Managed Objects for RS-232-like Hardware Devices using SMIv2
1769	Simple Network Time Protocol (SNTP)
2131, 2132	Dynamic Host Configuration; Protocol , DHCP Options and BOOTP Vendor Extensions
2570	Introduction to Version 3 of the Internet-standard Network Management Framework.
3164	The BSD Syslog Protocol

2. Standards Compliance for SNMP Traps

This section describes the standards compliance and any special operational features/options for the SNMP Traps supported by the GranDSLAM 3.0 products. The following traps are supported.

Table 1-4. Summary of Traps (1 of 7)

Trap	Description	VarBind(s)
Standard Traps		
LinkDown	Signifies that the unit has recognized a failure in one of the communication interfaces.	ifIndex(RFC 2863) ifAdminStatus(RFC 2863) ifOperStatus(RFC 2863) devLastTrapString
LinkUp	Signifies that the unit recognized that one of the communication interfaces has come up.	ifIndex(RFC 2863) ifAdminStatus(RFC 2863) ifOperStatus(RFC 2863) devLastTrapString
WarmStart	Signifies that the SNMPv2 entity, acting in an agent role, is reinitializing itself such that its configuration is unaltered.	devLastTrapString
AuthenticationFailureTrap	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 will be generated.	devLastTrapString
DSX Traps		
dsx1LineStatusChange	value of an instance dsx1LineStatus changes	ifIndex, dsx1LineStatus, dsx1LineStatusLastChange devLastTrapString
ATM-M4 Traps		
atmfM4IfAisAlarm	AIS condition has occurred.	ifIndex, ifOperStatus, atmfM4TrapAlarmSeverity devLastTrapString
atmfM4IfLcdAlarm	LCD (Loss of Cell Delineation) condition has occurred.	
atmfM4IfLosAlarm	LOS (Loss of Signal) condition has occurred.	

Table 1-4. Summary of Traps (2 of 7)

Trap	Description	VarBind(s)
atmfM4IfLofAlarm	LOF (Loss of Frame) condition has occurred.	ifIndex, ifOperStatus, atmfM4TrapAlarmSeverity devLastTrapString
atmfM4VclXConnCreated	Indicates that the VCL cross-connection has just been created. When a cross-connection is created along with its VCL endpoints, this trap should be emitted to indicate the creation of the entire group of objects.	atmVcCrossConnectL2HOperStatus, atmVcCrossConnectH2LOperStatus, devLastTrapString
atmfM4VclXConnDeleted	Indicates that the VCL cross-connection has just been created. When a cross-connection is created along with its VCL endpoints, this trap should be emitted to indicate the creation of the entire group of objects.	
atmfM4VclXConnChanged	Indicates that the VCL cross-connection configuration has been changed.	
atmfM4HwFanFailAlarm	This trap is generated when a cooling fan failure condition has occurred on the hardware unit associated.	entPhysicalIndex, ifOperStatus, atmfM4TrapAlarmSeverity devLastTrapString
Entity MIB Traps:		
entConfigChange	This trap is generated when the value of entLastChangeTime changes. It can be utilized by a NMS to trigger logical/physical entity table maintenance polls.	devLastTrapString
SHDSL MIB Traps:		
hdsl2ShdslLoopAttenCrossing	This notification indicates that the loop attenuation threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurrAtn, hdsl2ShdslEndpointThreshLoopAttenuation devLastTrapString
hdsl2ShdslSNRMarginCrossing	This notification indicates that the SNR margin threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurrSnrMgn, hdsl2ShdslEndpointThreshSNRMargin devLastTrapString
hdsl2ShdslPerfESThresh	This notification indicates that the errored seconds threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurr15MinES, hdsl2ShdslEndpointThreshES devLastTrapString

Table 1-4. Summary of Traps (3 of 7)

Trap	Description	VarBind(s)
hdsl2ShdslPerfSESThresh	This notification indicates that the severely errored seconds threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurr15MinSES, hdsl2ShdslEndpointThreshSES devLastTrapString
hdsl2ShdslPerfCRCanomaliesThresh	This notification indicates that the CRC anomalies threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurr15MinCRCanomalies, hdsl2ShdslEndpointThreshCRCanomalies devLastTrapString
hdsl2ShdslPerfLOSWSThresh	This notification indicates that the LOSW seconds threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurr15MinLOSWS, hdsl2ShdslEndpointThreshLOSWS devLastTrapString
hdsl2ShdslPerfUASThresh	This notification indicates that the unavailable seconds threshold has been reached/exceeded for the HDSL2/SHDSL segment endpoint.	hdsl2ShdslEndpointCurr15MinUAS, hdsl2ShdslEndpointThreshUAS devLastTrapString
hdsl2ShdslSpanInvalidNumRepeaters	This notification indicates that a mismatch has been detected between the number of repeater/regenerator units configuredfor a HDSL2/SHDSL line via the hdsl2ShdslSpanConfNumRepeaters object and the actual number of repeater/regenerator units discovered via the EOC.	hdsl2ShdslSpanConfNumRepeaters devLastTrapString
hdsl2ShdslLoopbackFailure	This notification indicates that an endpoint maintenance loopback command failed for an HDSL2/SHDSL segment.	hdsl2ShdslMaintLoopbackConfig devLastTrapString
hdsl2ShdslpowerBackoff	This notification indicates that the bit setting for powerBackoff in the hdsl2ShdslEndpointCurrStatus object for this endpoint has changed.	hdsl2ShdslEndpointCurrStatus devLastTrapString
hdsl2ShdsldeviceFault	This notification indicates that the bit setting for deviceFault in the hdsl2ShdslEndpointCurrStatus object for this endpoint has changed.	hdsl2ShdslEndpointCurrStatus devLastTrapString
hdsl2ShdsldcContinuityFault	This notification indicates that the bit setting for dcContinuityFault in the hdsl2ShdslEndpointCurrStatus object for this endpoint has changed.	hdsl2ShdslEndpointCurrStatus devLastTrapString
hdsl2ShdslconfigInitFailure	This notification indicates that the bit setting for configInitFailure in the hdsl2ShdslEndpointCurrStatus object for this endpoint has changed.	hdsl2ShdslEndpointCurrStatus devLastTrapString

Table 1-4. Summary of Traps (4 of 7)

Trap	Description	VarBind(s)
hdsl2ShdslprotocolInitFailure	This notification indicates that the bit setting forprotocolInitFailure in the hdsl2ShdslEndpointCurrStatus object for this endpoint has changed.	hdsl2ShdslEndpointCurrStatus devLastTrapString
hdsl2ShdslnoNeighborPresent	This notification indicates that the bit setting for noNeighborPresent in the hdsl2ShdslEndpointCurrStatus object for this endpoint has changed.	hdsl2ShdslEndpointCurrStatus devLastTrapString
hdsl2ShdslLocalPowerLoss	This notification indicates timpending unit failure due to loss of local power (last gasp).	hdsl2ShdslInvVendorID devLastTrapString
SONET LINEAR APS MIB Traps:		
apsEventSwitchover	This notification is sent when the value of an instance of apsChanStatusSwitchovers increments	apsChanStatusSwitchovers, apsChanStatusCurrent devLastTrapString
apsEventModeMismatch	This notification is sent when the value of an instance of apsStatusModeMismatches increments	apsStatusModeMismatches, apsStatusCurrent devLastTrapString
apsEventChannelMismatch	This notification is sent when the value of an instance of apsStatusChannelMismatches increments	apsStatusChannelMismatches, apsStatusCurrent devLastTrapString
apsEventPSBF	This notification is sent when the value of an instance of apsStatusPSBFs increments	apsStatusPSBFs, apsStatusCurrent devLastTrapString
apsEventFEPLF	This notification is sent when the value of an instance of apsStatusFEPLFs increments	apsStatusFEPLFs, apsStatusCurrent devLastTrapString
Enterprise Specific Traps:		
mpeSelfTestFailure	A hardware failure of the unit was detected as part of the unit's selftest. This trap will be generated after the unit has completed initialization	entPhysicalIndex mpedevSelfTestResults devLastTrapString
mpeCcn	This trap is generated to indicate a change in the configuration in the entity unit specified by the entPhysicalIndex value.	entPhysicalIndex devLastTrapString

Table 1-4. Summary of Traps (5 of 7)

Trap	Description	VarBind(s)
devSonetStatusChange	A devSonetStatusChange trap is sent when the value of an instance of sonetSectionCurrentStatus or sonetLineCurrentStatus or sonetPathCurrentStatus changes.	devSonetStatusLastChange sonetSectionCurrentStatus, sonetLineCurrentStatus, sonetPathCurrentStatus devLastTrapString
pdnRedunEventNoActiveModule	This notification will be issued when no module is the Active State.	pdnRedunAlarmStatus devLastTrapString
pdnRedunEventHwIncompatible	This notification will be issued when the modules in a redundancy configuration present hardware incompatibilities.	pdnRedunAlarmStatus devLastTrapString
pdnRedunEventStandbyAlarmOrReset	This notification will be issued when one of the modules configured for redundancy is either in the Standby/Alarm or Reset Test state.	pdnRedunAlarmStatus devLastTrapString
pdnRedunEventFwIncompatible	This notification will be issued when the modules in a redundancy configuration present firmware incompatibilities.	pdnRedunAlarmStatus devLastTrapString
pdnRedunEventCfgIncompatible	This notification will be issued when the modules in a redundancy configuration present configuration incompatibilities.	pdnRedunAlarmStatus devLastTrapString
ADSL Specific Traps:		
adslAtucPerfLofsThreshTrap	Loss of Framing 15-minute interval threshold reached	ifIndex adslAturPerfCurr15MinLofs adslAturThresh15MinLofs devLastTrapString
adslAtucPerfLossThreshTrap	Loss of Signal 15-minute interval threshold reached	ifIndex adslAtucPerfCurr15MinLoss adslAtucThresh15MinLoss devLastTrapString
adslAtucPerfESsThreshTrap	Errored Second 15-minute interval threshold reached	ifIndex adslAtucPerfCurr15MinESs adslAtucThresh15MinESs devLastTrapString
adslAtucRateChangeTrap	The ATUCs transmit rate has changed (RADSL mode only)	ifIndex adslAtucChanCurrTxRate adslAtucChanPrevTxRatedevLastTrapString

Table 1-4. Summary of Traps (6 of 7)

Trap	Description	VarBind(s)
adslAtucPerfLolsThreshTrap	Loss of Link 15-minute interval threshold reached	ifIndex adslAtucPerfCurr15MinLols adslAtucThresh15MinLols devLastTrapString
adslAtucInitFailureTrap	ATUC initialization failed. See adslAtucCurrStatus for potential reasons	ifIndex adslAtucCurrStatus devLastTrapString
adslAturPerfLofsThreshTrap	Loss of Framing 15-minute interval threshold reached	ifIndex adslAturPerfCurr15MinESs adslAturThresh15MinESs devLastTrapString
adslAturPerfLossThreshTrap	Loss of Signal 15-minute interval threshold reached	ifIndex adslAturPerfCurr15MinLoss adslAturThresh15MinLoss devLastTrapString
adslAturPerfLprsThreshTrap	Loss of Power 15-minute interval threshold reached	ifIndex adslAturPerfCurr15MinLprs adslAturThresh15MinLprs devLastTrapString
adslAturPerfESsThreshTrap	Errored Second 15-minute interval threshold reached	ifIndex adslAturPerfCurr15MinESs adslAturThresh15MinESs devLastTrapString
adslAturRateChangeTrap	The ATURs transmit rate has changed	ifIndex adslAturChanCurrTxRate adslAturChanPrevTxRate devLastTrapString
adslAtucSesLThreshTrap	Severly errored seconds 15 minutes threshold reached	ifIndex adslAtucPerfCurr15MinSesL devLastTrapString
adslAtucUasLThreshTrap	Unavailable seconds 15 minutes threshold reached	ifIndex adslAtucPerfCurr15MinUasL devLastTrapString

Table 1-4. Summary of Traps (7 of 7)

Trap	Description	VarBind(s)
adslAturSesLThreshTrap	Severly errored seconds 15 minutes threshold reached	ifIndex adslAturPerfCurr15MinSesL devLastTrapString
adslAturUasLThreshTrap	Unavailable seconds 15 minutes threshold reached	ifIndex adslAturPerfCurr15MinUasL devLastTrapString
IMA Traps		
imaFailureAlarm	The imaFailureAlarm provides a method for an agent implementing IMA to notify an NMS of an alarm condition.	ifIndex, imaAlarmStatus, imaAlarmType devLastTrapString

3. Standards Compliance to SNMP MIBs

This section describes the standards compliance and any special operational features/options for the SNMP MIBs supported on the GranDSLAM 3.0 product. The term NO_SUCH_NAME_ERROR will be used to say that either:

a) a whole table does not exist in the MIB tree or

b) a particular object does not exist in the MIB tree

Mandatory groups will be supported as is. If a value can not be determined for an mandatory object, a default value (defined in the MIBWG guidelines) will be assigned and will be specified in this document. However, this SNMP agent is not required to support optional groups. The agent will return the value NO_SUCH_NAME_ERROR for a get request of the objects contained in any optional group not supported. Network Management Systems must be able to handle this situation better known as sparse tables. An exception to the later is the ifTable. The ifTable will not be a sparse table and will be fully supported.

When a particular enumeration for a particular object is not available, attempt to write it by a management entity will result in a failure of BAD_VALUE or GENERAL_ERROR.

The following diagram may be useful in visualizing the interfaces on the product.

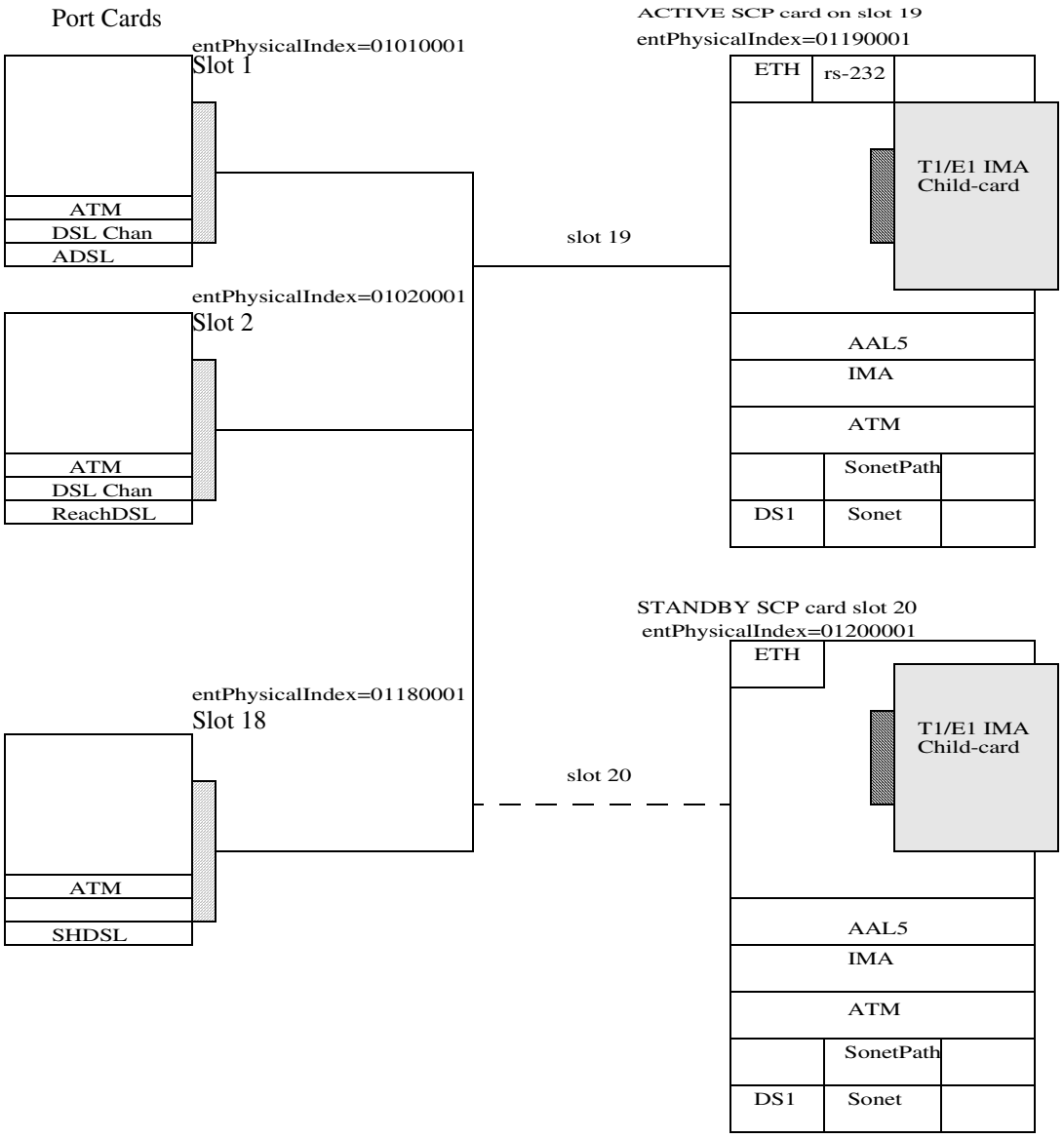

Figure 1-1. GranDSLAM Interfaces

3.1 Management Information Base (MIB)-II - RFC 1213.

The objects defined by MIB-II are organized into 10 different groups. The GranDSLAM 3.0 will implement only those groups where the semantics of the group are applicable to the implementation of the stack (for example, if EGP is not implemented the EGP group will not be supported). The MIB-II object groups supported by the GranDSLAM 3.0 are as follows:

System Group	Supported.
Interfaces Group	Supported for the Internet Assigned Numbers Authority (IANA) ifTypes: ATM Cell layer (IANAifType = 37) ATM AAL-5 layer (IANAifType = 49) DS-1 Interface (IANAifType = 18) for T1 interface. E1 Interface (IANAifType = 19) for E1 interface. ADSL Interface (IANAifType=94) SHDSL Interface (IANAifType=169) ETHERNET Interface (IANAifType = 6) ADSL Fast Channel (IANAifType = 125), or ADSL Interleave Channel (IANAifType = 124) IMA Interface (IANAifType = 107) SONET Interface (IANAifType = 39) SONETPATH (IANAifType = 50)
IP Group	NOT_SUPPORTED.
ICMP Group	NOT_SUPPORTED.
TCP Group	NOT_SUPPORTED.
UDP Group	NOT_SUPPORTED.
EGP Group	NOT_SUPPORTED
Transmission Group	SUPPORTED
SNMP Group	SUPPORTED.

3.2 System Group, MIB-II (RFC 1213)

The System Group Objects are fully supported by the GranDSLAM 3.0. The following sections provide clarification for objects contained in the System Group when it is not clear how the object definition in MIB-II is related to the GranDSLAM 3.0.

3.2.1 "sysDescr" Object (system 1)

This object provides the full name and version identification for the system's hardware and software. This object will be set to display the following string:

"Paradyne DSLAM *Chassis*;"

Where:

Chassis This value will be 8820 or 8620.

3.2.2 "sysObjectID" Object (system 2)

This object provides the authoritative identification of the network management subsystem contained in the unit. This object will be set to display the following object identifier:

sme-chassis-8820[Company Enterprise OID].1.14.9.14.5.1

sme-chassis-8620[Company Enterprise OID].1.14.9.14.5.2

3.2.3 "sysUpTime" Object (system 3)

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

3.2.4 "sysContact" Object (system 4)

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.

3.2.5 "sysName" Object (system 5)

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.

3.2.6 "sysLocation" Object (system 6)

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

3.2.7 "sysServices" Object (system 7).

This object provides a value which indicates the set of services that are potentially offered by the DSL unit. The value is a sum. This sum initially takes the value zero. Then, for each layer L in the range of 1-7 (for 7 OSI layers) that this node performs transactions for, 2 raised to L-1 is added to the sum. For example, a node performing only routing functions would have a value of 4(z^(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.

The following table shows the values supported :

Table 1-5. sysServices

Layer (L)	Functionality	Value(2^(L-1)	Sum
1	Physical	2^(1-1) = 2^0 = 1	1
2	DataLink/Subnetwork	2^(2-1) = 2^1 = 2	3
3	Internet	2^(3-1) = 2^2 = 4	7
4	End-to-End	2^(4-1) = 2^3 = 8	15
5	Session	2^(5-1) = 2^4 = 16	31
6	Presentation	2^(6-1) = 2^5 = 32	63
7	Application	2^(7-1) = 2^6 = 64	127

Therefore this object will be set to the value of 127.

4. Evolution of the Interfaces Group of MIB-II (RFC 2863)

The interfaces group consists of an object indicating the number of interfaces supported by the unit and an interface table containing an entry for each interface in the GranDSLAM 3.0. Since RFC 2863 is an SNMPv2 MIB it will be converted to SNMPv1 for support by the GranDSLAM 3.0. There will be entries for WAN interfaces, ATM cell interfaces, an AAL5 interface, management interfaces and DSL interfaces.

4.1 "ifNumber" Object (interfaces 1)

ifNumber will be supported as specified in the Evolution MIB and will specify the number of rows in the ifTable for this particular device.

4.1.1 "ifIndex" Object (ifEntry 1)

This object provides the index into the ifTable and typically into tables in other MIBs also. A unique value, greater than zero, for each interface or interface sublayer in the managed system. It is recommended that values be assigned contiguously starting from 1. The value for each interface sublayer must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

4.1.1.1 ifIndex scheme

This section introduces the ifIndex scheme to be used in the GranDSLAM 3.0 product. It describes the interfaces to be managed as specified in the system requirements along with their corresponding ifIndex assigned. The assignment of the values to the ifIndex in the product is static and maintained across power cycles and software upgrades. The first column in Table 1-6, GranDSLAM 3.0 Interface List, is the Interface description. The second column corresponds to the layer in the stack where this interface fits (It does not necessary match the layer in the OSI model). The third column is the number of this interface supported in the current software release. The forth column is the maximum number of this interface that is envisioned in this product. The fifth column is the name associated with that instance of the interface.

The 'x' suffix in some ifName values corresponds to the interface/port number within the same chassis.

Table 1-6. GranDSLAM 3.0 Interface List

Interface	Layer	Supported	Maximum	ifName
Console port[a]	1	1	1	CONx
Ethernet Mgmt Port	1	2	2	ETHx
T1	1	8	8	T1x
E1	1	8	8	E1x
SONET	1	2	2	SONET_x
SONETPATH	2	2	2	SONETPATH_x
ATM Link[b]	3	8	8	ATMUP_x
IMA	4	1	8	IMAx
ATM AAL5	4	1	1	AAL5_x
DSL port	1	24*18	48*18	DSL_x
DSL channels	2	48*18	48*2*18=96*18	DSLChan_x
DSL ATM Links[c]	3	48*18	48*2*18=96*18	ATMx

[a] The console will not be listed in the interface group, but it will be in the entPhysicalTable of the entityMIB.

[b] ATM Link refers to the ATM layer supported over the T1/E1 or OC3 interfaces.

[c] DSL ATM Links refers to the ATM layer supported over the DSL interface.

The following table shows how the different interfaces stack in the product.

Table 1-7. Interface Stack

Description	Console	DSL Port		WAN Port	Management
Layer 5				AAL5	
Layer 4				IMA	
Layer 3		ATM		ATM	
Layer 2		DSL Channel		SonetPath	
Layer 1	rs232	ReachDSL, ADSL	SHDSL	T1/E1, Sonet	Ethernet

4.1.1.1.1 Interface indexes (ifIndex).

A unique value, greater than zero number will be assigned to each interface. The following ifIndex scheme was developed by a committee to solve a general stack hierarchy and it is shown in Figure 1-2:

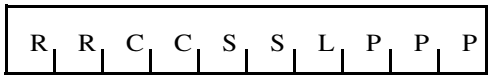

Figure 1-2. ifIndex Scheme

Where:

- RR: Reserved for future use.
- CC: Chassis number from 1..99. For the GranDSLAM 3.0, this value will always be 1, identifying the 8820 or the 8620 chassis.
- SS: Slot number from 1..99. For the GranDSLAM 3.0, this value will range from 1 through 20.
- Slot 00 will have a special meaning on the GranDSLAM 3.0. Slot 00 will be used for the interfaces contained in the SCP cards.
- L: Layer number associated in the interface stack in Table 1-7, Interface Stack.
- PPP: Port/Interface number from 1..999.

The value 0 will be used as a special case in any of the above fields to denote N/A.

The following table shows the assigned ifIndexes to be used for the GranDSLAM 3.0 product. In the table, the DSL ports are enumerated from 1-48, with the Layer field set to 1. The product will support 2 DSL Channels (Interleave, Fast) per DSL ports, that make a total of 96 channels numbered from 1-96 with the Layer field set to 2. Similarly, their corresponding DSL ATM links will be numbered from 1-96, but with the Layer field set to 3. In the case of the WAN ports, the number starts at 901 for the first T1/E1or OC3 link and increments up to the max number of WAN ports supported. When some of these DS1 are aggregated to form an IMA Group, the first IMA group starts at 910, but with the Layer field set to 4. Similarly, the ATM links associated with the DS1 port will carry the same starting number as the physical interface but with the Layer Field set to 3, which corresponds to the ATM layer in Table 1-7, Interface Stack. Finally, other ports dedicated to management of the device such as Ethernet ports and async ports will have ifIndexes starting at 801 for the first port, and so on.

The suffix ‘cc’ indicates the chassis number where the interface is instantiated. The chassis number for the 8820 or the 8620 will be 1 for the GranDSLAM 3.0.

The suffix ‘ss’ indicates the slot number where the interface is instantiated. For the GranDSLAM 3.0, this number corresponds to the physical position where the associated module is located in the 8820 or 8620 chassis.

For the GranDSLAM 3.0, the value ss=’00’ have a special meaning. A logical slot number of 00 will be used for those interfaces contained and accessible in the SCP cards. Therefore, these interfaces will be independent and remain contant regardless of which SCP is in charge. However, the internal software may require that the corresponding slot based ifIndexes in the SCP be also supported.

Table 1-8. ifIndex Assignments

Interface*	Interface Name (ifName)	Interface Type (IANAifType)	ifInterface Index (ifIndex)	Interface Descr (ifDescr)
Console port	CONx **	rs232 (33)	ccss1801	“Console port in unit ’cc’, slot ‘ss”
Ethernet Mgmt port	ETHx	ethernetCsmacd(6)	ccss1802	“Ethernet Management port in unit ’cc’, slot ‘ss”
T1	T1_x	ds1 (18)	ccss190x	“T1 interface in unit ‘cc’, slot ‘ss‘ ” Where x=[1..8]
E1	E1_x	e1 (19)	ccss190x	“E1 interface in unit ‘cc’, slot ‘ss’ ” Where x=[1..8]
OC3	SONET_x	sonet(39)	ccss190x	"Sonet interface in unit ‘cc’, slot ‘ss’ ” Where x=[1..2]
SONETPATH	SONETPATH_x	sonetPath(50)	ccss290x	"SonetPath interface in unit ‘cc’, slot ‘ss’ ” Where x=[1..2]
ATM Link	ATMUP_x	atm (37)	ccss390x	“ATM Link in unit ‘cc’, slot ‘ss ” Where x=[1..8]
IMA	IMA_x	atmIma(107)	ccss490x	“IMA interface in unit ‘cc’, slot ‘ss’ ” Where x=[0..7]
AAL5	AAL5_x	aal5 (49)	ccss5901	“ATM-AAL5 interface in unit ‘cc’, slot ‘ss’ ”
DSL port	DSL_xx	adsl (94)	ccss10xx	“DSL port ‘xx’in unit ‘cc’ , slot ‘ss’” Where xx=[1..48]
SHDSL port	SHDSL_xx	shdsl (169)	ccss10xx	“SHDSL port ‘xx’ in unit ‘cc’ , slot ‘ss’” Where xx=[1..48]
DSL Channels	DSLChan_xx	fast (125)/ interleave(124)	ccss20xx	“DSL Fast Channel ‘xx’in unit ‘cc’, slot ‘ss’ ” Where xx=[1..48] are assigned to fast channels and xx=[49..96] are assigned to interleave channels.
ATM	ATMx	atm (37)	ccss30xx	“DSL ATM Port ‘xx’ in unit ‘cc’, slot ‘ss’ ” Where xx=[1..96]

* Please refer to Table 1-6, GranDSLAM 3.0 Interface List, for the exact number of interfaces supported of each class in the current release.

**Console interface will not be displayed in the ifTable. Only in the entPhysicalTable.

4.1.2 “ifDescr” Object (ifEntry 2)

This object provides the textual information about the interface. Each interface will be set to a text string as specified in Table 1-8, ifIndex Assignments.

4.1.3 "ifType" Object (ifEntry 3)

This object identifies the type of the interface based on the physical/link protocol(s). The IfType values in Table 1-8, ifIndex Assignments are supported.

4.1.4 "ifMtu" Object (ifEntry 4)

This object identifies the largest datagram, associated with a management link, which can be sent/received on the interface. ifMTU will have the following values:.

Table 1-9. ifMtu

Interface	ifMtu
AAL5	4128*
ATMUP	0
DSL	0
DSLATM	0
DSLChan	0
E1	0
ETH	1500
IMA	0
SONET	0
SONETPath	0
T1	0

* Per RFC 2515, this object is set to the largest PDU size for the AAL5 CPCS layer that can be processed by the AAL5 Entity.

4.1.5 "ifSpeed" Object (ifEntry 5)

This object provides the interfaces's current bandwidth in bits per second. The value of this object for each interface will be specified as follows:

Table 1-10. ifSpeed

Interface	ifSpeed
AAL5	0
ATMUP	155000000
DSL	16000000
DSLATM	16000000
DSLChan	16000000
E1	2048000
ETH	100000000
IMA	[1544000..8*2048000]
SONET	155000000
SONETPath	155000000
T1	1544000

4.1.6 "ifPhysAddress" Object (ifEntry 6)

This object is used to specify the interface's address at its protocol sub-layer. The interface's media-specific MIB must define the bit and byte ordering and format of the value contained by this object. For the GranDSLAM 3.0 interfaces, this object should only contain the ethernet address (MAC Address) of the unit. All other interfaces are in blank.

4.1.7 "ifAdminStatus" Object (ifEntry 7)

This object is used to specify the desired state (configuration) of the interface. When the GranDSLAM 3.0 initializes, all physical interfaces start with ifAdminStatus in the up(1) state. As a result of either explicit management action or per configuration ifAdminStatus is changed to down(2). The ifAdminStatus values which may be assigned to each interface after initialization are listed in the following table:

Table 1-11. ifAdminStatus

Interface	ifAdminStatus
AAL5	Up(1)
ATMUP	Up(1)
DSL	Up(1), Down(2)
DSLATM	Up(1)
DSLChan	Up(1)
E1	Up(1), Down(2)
ETH1	Up(1), Down(2)
IMA	Up(1), Down(2)
SONET	Up(1), Down(2)
SONETPath	Up(1), Down(2)
T1	Up(1), Down(2)

4.1.8 "ifOperStatus" Object (ifEntry 8)

This object is used to specify the current operational state of the interface. The value of this object for each interface will be defined as follows:

Table 1-12. ifOperStatus

Interface	ifOperStatus
AAL5	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
ATMUP	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
DSL	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
DSLATM	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
DSLChan	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
E1	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
ETH1	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
IMA	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
SONET	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
SONETPath	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)
T1	Up(1), Down(2), Testing(3), Unknown(4), notPresent(6), lowerLayerDown(7)

4.1.9 "ifLastChange" Object (ifEntry 9)

This object contains the value of "sysUpTime" at the time the interface entered its current operational state.

4.1.10 ATM and ATM Cell Management Input and Output counter explanations:

- ifInOctets – The number of received octets over the interface, that is, the number of received, assigned cells, multiplied by 53.
- ifOutOctets – The number of transmitted octets over the interface,that is, the number of transmitted, assigned cells, multiplied by 53.
- ifInErrors – The number of cells dropped due to uncorrectable HEC errors.
- ifInUnknownProtos – The number of received cells discarded during cell header validation, and cells with invalid cell header patterns. If cells with undefined PTI values are discarded, they are also counted here. (NOT_SUPPORTED)

4.1.11 Input Counters (objects ifEntry 10 to ifEntry 15)

4.1.11.1 "ifInOctets" Object (ifEntry 10)

Table 1-13. ifInOctets

Interface	ifInOctets
AAL5	Number of received AAL5 CPCS PDU octets
ATMUP	Number of Cells received Multiplied by 53
DSL	0
DSLATM	Number of Cells received Multiplied by 53
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1	0

4.1.11.2 "ifInUcastPkts" Object (ifEntry 11)

Table 1-14. ifInUncastPkts

Interface	ifInUcastPkts
AAL5	PDUs inbound:The number of AAL5 CPCS PDUs passed to a higher layer
ATMUP	0
DSL	0
DSLATM	0
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1	0

4.1.11.3 "ifInUncastPkts" Object (ifEntry 12)

Table 1-15. ifInUncastPkts

Interface	ifInUncastPkts
DSL	0
DSLChan	0
DSLATM	0
T1/E1	0
E1	0
IMA	0
ATMUP	0
AAL5	0
ETH1	0
SONET	0
SONETPath	0

4.1.11.4 "ifInDiscards" Object (ifEntry 13)

Table 1-16. ifInDiscards

Interface	ifInDiscards
AAL5	Number of AAL5 CPCS PDUs discarded
ATMUP	0
DSL	Number of Octets received for discard
DSLATM	0
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.11.5 "ifInErrors" Object (ifEntry 14)

Table 1-17. ifInErrors

Interface	ifInErrors
AAL5	Number of errored AAL5 CPCS PDUs received. These include CRC-32 errors, SAR Time-out error, and oversized DSU errors
ATMUP	0
DSL	0
DSLATM	Number of cells dropped due to uncorrectable HEC
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.11.6 "ifInUnknownProtos" Object (ifEntry 15)

Table 1-18. ifInUnknownProtos

Interface	ifInUnknownProtos
AAL5	0
ATMUP	0
DSL	0
DSLATM	Number of cells dropped during header validation, and cells with unrecognized header patterns
DSLChan	0
E1	0
ETH1	PDUs of incorrect type
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.12 Output Counters (objects ifEntry 16 to ifEntry 20)

4.1.12.1 "ifOutOctets" Object (ifEntry 16)

Table 1-19. ifOutOctets

Interface	ifOutOctets
AAL5	Number of Octets in Pkts
ATMUP	Number of cells sent multiplied by 53
DSL	Number of Octets sent
DSLATM	Number of cells sent multiplied by 53
DSLChan	Number of Octets sent
E1	0
ETH1	0
IMA	Number of Octets sent
SONET	0
SONETPath	0
T1	0

4.1.12.2 "ifOutUcastPkts" Object (ifEntry 17)

Table 1-20. ifOutUcastPkts

Interface	ifOutUcastPkts
AAL5	PDUs outbound: The number of AAL5 CPCS PDUs received from a higher layer for transmission.
ATMUP	0
DSL	0
DSLATM	0
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.12.3 "ifOutNUcasts" Object (ifEntry 18)

Table 1-21. ifOutNUcasts

Interface	ifOutNUcasts
AAL5	0
ATMUP	0
DSL	0
DSLATM	0
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.12.4 "ifOutDiscards" Object (ifEntry 19)

Table 1-22. ifOutDiscards

Interface	ifOutDiscards
AAL5	0
ATMUP	0
DSL	0
DSLATM	0
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.12.5 "ifOutErrors" Object (ifEntry 20)

Table 1-23. ifOutErrors

Interface	ifOutErrors
AAL5	0
ATMUP	0
DSL	0
DSLATM	0
DSLChan	0
E1	0
ETH1	0
IMA	0
SONET	0
SONETPath	0
T1/E1	0

4.1.13 "ifOutQLen" (ifEntry 21)

This object contains the length of the output packet queue. It will contain a value assigned by the unit which is used for each interface.This object is not supported and will return a huge invalid number.

4.1.14 "ifSpecific" (ifEntry 22)

This object contains a reference to MIB definitions specific to the particular media being used.

This object is not supported and will return a value of 0.0 (that is, no MIB definition specific to the particular media is available).

4.2 Extension to the Intf. Table (ifXTable) - (RFC 2863)

The Extension to the interface table contains additional objects for the interface table. Only the following objects in the Extension to the Interface Table will be supported in the GranDSLAM 3.0:

Table 1-24. Extension to the Interface Table

Interface	ifInMulticastPkts(2) (Read Only)	ifInBroadcastPkts(3) (Read Only)	ifOutMulticastPkts(4) (Read Only)	ifOutBroadcastPkts(5) (Read Only)	ifHCInOctets(6) (Read Only)	ifHCInUcastPkts(7) (Read Only)	ifHCInMulticastPkts(8) (Read Only)	ifHCInBroadcastPkts(9) (Read Only)	ifHCOutOctets(10) (Read Only)	ifHCOutUcastPkts(11) (Read Only)	ifHCOutMulticastPkts(12) (Read Only)	ifHCOutBroadcastPkts(13) (Read Only)	ifLinkUpDownTrapEnable(14) (Read Write)	ifHighSpeed(15) (Read Only)	ifPromiscuousMode(16) (Read Write)	ifConnectorPresent(17) (Read Only)
AAL5	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	Z	Z	S
ATMUP	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	Z	Z	S
DSL	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	S	Z	S
DSLATM	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	S	Z	S
DSLChan	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	S	Z	S
E1	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	Z	Z	S
ETH1	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	Z	S	S
IMAx	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	Z	Z	S
SONET	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	S	Z	S
SONETPath	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	S	Z	S
T1	Z	Z	Z	Z	U	U	U	U	U	U	U	U	S	Z	Z	S
Legend: S =Supported, Z=returns zero, U=Unsupported (returns no-such-name) Note: For ATM Cell AAL5 Sublayers, see also RFC 2515																

4.2.1 "ifLinkUpDownTrapEnable" Object (ifXEntry 14)

This object is used to indicate whether linkUp/linkDown or enterpriseSpecific traps should be generated for the interface. This object is supported read-only.

4.3 Interface Stack Group - (RFC 2863)

The interface stack group consists of an object containing information on the relationships between the multiple sub-layers of network interfaces. In the GranDSLAM 3.0, the interface stack group will be used to show the relationship between the multiple logical and physical interfaces, if a physical interface exists.

The ifStackTable entries for the product is illustrated below, please refer to Table 1-6, GranDSLAM 3.0 Interface List, for the exact number of interfaces of each class supported in the current release:

Table 1-25. ifStackTable

GranDSLAM 3.0 Higher Interface Number	Mid-InterfaceNumber	Lower Interface Number
AAL5_x	IMA_x	ATMUP_x *
AAL5_x	ATMUP_x	SONETPATH_x **
ATMUP_x	SONETPATH_x	SONET_x
DSLATM_x	DSLChan_x	DSL_x
IMA_x	ATMUP_x	T1/E1_x

** If the associated IMA interface is present*

*** If the associated SONET interface is present*

4.3.1 "ifStackHigherLayer" Object (ifStackEntry 1)

This object provides the ifIndex corresponding to the higher sub-layer running on 'top of' the interface specified in ifStackLowerLayer.

Table 1-26. ifStackHigherLayer

GranDSLAM 3.0 Higher Interface Number	GranDSLAM 3.0 Lower Interface Number	Comments
ccss10xx	0	DSL Ports
ccss20xx	ccss10xx	DSL Channels over DSL
ccss30xx	ccss20xx	DSL ATM over DSL Channels
ccss1802	0	Ethernet Ports
ccss190x	0	UPLINK Ports
ccss390x	ccss190x	ATM over UPLINK ports
ccss490x	ccss390x	IMA over ATM UpLinks***
ccss5901	ccss490x/ccss390x	AAL5

*** If the associated ATM link is part of an IMA group

4.3.2 "ifStackLowerLayer" Object (ifStackEntry 2)

This object provides the ifIndex corresponding to the lower sub-layer running on 'below' the interface specified in the corresponding ifStackHigherLayer.

Table 1-27. ifStackLowerLayer

GranDSLAM 3.0 Higher Interface Number	GranDSLAM 3.0 Lower Interface Number	Comments
0	ccss30xx	Nothing over DSL ATM Links
ccss30yy	ccss20yy	ATM over DSL channels
ccss20yy	ccss10yy	DSL channels over DSL
0	ccss1802	Nothing over Ethernet
0	ccss5901	Nothing over AAL5
ccss5901	ccss490y	AAL5 over IMA (IMA)
ccss490x	ccss390x	IMA over ATM Uplink (IMA)
ccss5901	ccss390y	AAL5 over ATM Uplink (SONET)
ccss390y	ccss190y	ATM over UPLINK ports

** If the associated ATM link is part of an IMA group*

*** If the associated ATM link is not part of an IMA group*

4.3.3 "ifStackStatus" Object (ifStackEntry 3)

This object **is supported as a "read only" variable**. Only active(1) and notInService(2) are supported.

4.4 Transmission Group, MIB-II (RFC 1213) - (SUPPORTED)

The Transmission Group Objects is SUPPORTED.

4.5 SNMP Group, MIB-II (RFC 1213)

The SNMP Group Objects that apply to a management agent are fully supported. The following objects apply only to an NMS and return a zero value if accessed.

- snmpInBadCommunityName (snmp 4)
- snmpInTooBigs (snmp 8)
- snmpInNoSuchNames (snmp 9)
- snmpInBadValues (snmp 10)
- snmpInReadOnlys (snmp 11)
- snmpInGenErrs (snmp 12)
- snmpOutGetRequests (snmp 25)
- snmpOutGetNexts (snmp 26)
- snmpOutSetRequests (snmp 27)

The following objects are supported:

- snmpInPkts (snmp 1)
- snmpOutPkts (snmp 2)
- snmpInBadVersions (snmp 3)
- snmpInBadCommunityUses (snmp 5)
- snmpInASNParseErrs (snmp 6)
- snmpInTotalReqVars (snmp 13)
- snmpInTotalSetVars (snmp 14)
- snmpInGetRequests (snmp 15)
- snmpInGetNexts (snmp 16)
- snmpInSetRequests (snmp 17)
- snmpInGetResponses (snmp 18)
- snmpInTraps (snmp19)
- snmpOutTooBigs (snmp 20)
- snmpOutNoSuchNames (snmp 21)
- snmpOutBadValues (snmp 21)
- snmpOutGenErrs (snmp 24)
- snmpOutGetResponses (snmp 28)
- snmpOutTraps (snmp 29)
- snmpEnableAuthenTraps (snmp 30) (Read/Write)

5. Physical Layer

5.1 ADSL MIB (RFC 2662)

Table 1-28. ADSL MIB

Table	Comments
adslLineEntry	Supported as is
adslAtucPhysEntry	Supported as is
adslAturPhysEntry	Supported as is
adslAtucChanEntry	Supported as is
adslAturChanEntry	Supported as is
adslAtucPerfDataEntry	Supported as is
adslAturPerfDataEntry	Supported as is
adslAtucIntervalEntry	Supported as is
adslAtucIntervalEntry	Supported as is
adslChanIntervalEntry	Supported as is
adslChanIntervalEntry	Supported as is
adslLineConfProfileEntry	Supported as is
adslLineAlarmConfProfileEntry	Supported as is

5.2 ADSL-EXT MIB (draft-ietf-adslmib-adslext-xx.txt)

Table 1-29. ADSL-EXT MIB

Table	Comments
adslLineExtEntry	Supported as is
adslAtucPerfDataExtEntry	Supported as is
adslAturPerfDataExtEntry	Supported as is
adslConfProfileExtEntry	Supported as is
adslAlarmConfProfileExtEntry	Supported as is

5.3 Paradyne ReachDSL MIB (pdn_reachDSL.mib)

This Paradyne enterprise MIB provides objects for configuration of objects specific to ReachDSL devices as well as for Spectrum Management objects specific to ReachDSL products. The various objects supported are shown in the following table.

Table 1-30. ReachDSL MIB

Table	Comments
reachDSLSpectrumMgmtSelection	Supported as is
reachDSLSpectrumMgmtConfEntry	Supported as is
reachDSLSpectrumMgmtLoopMeasurementMethod	Supported as is
reachDSLSpectrumMgmtEWLUnits	Supported as is
reachDSLSpectrumMgmtMode	Supported as is
reachDSLSpectrumMgmtLineInfoEntry	Supported as is
reachDSLLineEntry	Supported except for reachDSLCircuitIdentifier.

The ATU-C and ATU-R Tx Rates configurable for a specific Spectrum Zone and EWL are dependent on the spectrum management guidelines for that specific zone and can be obtained from the ReachDSL Core specifications as well as from the System Requirements documents.

5.4 DS1 MIB (RFC 2495)

Table 1-31. DS1 MIB

Table	Comments
dsx1ConfigEntry	Supported as is
dsx1CurrentEntry	Supported as is
dsx1IntervalEntry	Supported as is
dsx1TotalEntry	Supported as is
dsx1ChanMappingEntry	Supported as is
dsx1FarEndCurrentEntry	Supported as is
dsx1FarEndIntervalEntry	Supported as is
dsx1FarEndTotalEntry	Supported as is

5.5 Paradyne DS1-EXT MIB (PDN-DS1EXT-MIB).

This Paradyne MIB is an extension to the standard DS1 MIB (RFC 2495). This MIB provides additional DS1 and G.703 (E1) configuration objects not provided by RFC 2495.

The various objects supported are shown in the following table.

Table 1-32. DS1-EXT MIB

Table	Comments
pdnDs1ExtConfTable	Supported objects: pdnDs1ExtConfLineLengthType pdnDs1ExtConfLineLength pdnDs1ExtConfLineBuildOut pdnDs1ExtConfConnector

5.5.1 pdnDs1ExtConfLineLengthType (RW):

This entry specifies the type of loop length for the interface. The posibble values are shortHaul(1), longHaul(2). short-haul is intended for intra-building use. long-haul is intended for inter-building use. **The default value is long-haul.**

5.5.2 pdnDs1ExtConfLineLength (RW):

This entry specifies the loop length, in feet, for a short-haul DS1 line. This object only applies to DS1 interfaces. Note: This object shares a mutually exclusive relationship with the pdnDs1ExtConfLineBuildOut object. Only one of the two can be used for configuration at a time, based on the pdnDs1ExtConfLineLengthType object. **The default value is feet000To133.**

5.5.3 pdnDs1ExtConfLineBuildOut (RW):

This entry specifies the line build out, in decibels, for a long-haul DS1 line. The possible values are: dB0Pnt0(1), dB7Pnt5(2), dB15Pnt0(3), dB22Pnt5(4). **The default value is dB0Pnt0(1)**. This object only applies to DS1 interfaces.

5.5.4 pdnDs1ExtConfConnector (RW):

This entry specifies the type of connector to be used. This object only applies to G.703 interfaces. The possible values are: bnc(1), rj48(2). **The default value is rj48(2)**.

5.6 SONET-MIB (RFC 2558)

Table 1-33. SONET-MIB

Table	Comments
sonetMediumEntry	Some objects are supported read-only
sonetSectionCurrentEntry	Supported as is
sonetSectionIntervalEntry	Supported as is
sonetLineCurrentEntry	Supported as is
sonetLineIntervalEntry	Supported as is
sonetFarEndLineCurrentEntry	Supported as is
sonetFarEndLineIntervalEntry	Supported as is
sonetPathCurrentEntry	Supported as is
sonetPathIntervalEntry	Supported as is
sonetFarEndPathCurrentEntry	Supported as is
sonetFarEndPathIntervalEntry	Supported as is

5.6.1 sonetMediumEntry

Table 1-34. sonetMediumEntry

Table	Comments
sonetMediumType	Supported read-write
sonetMediumTimeElapsed	Supported read-only
sonetMediumValidIntervals	Supported read-only
sonetMediumLineCoding	Supported read-only
sonetMediumLineType	Supported read-only
sonetMediumCircuitIdentifier	Supported read-write
sonetMediumInvalidIntervals	Supported read-only
sonetMediumLoopbackConfig	Supported read-write

5.6.2 sonetSESthresholdSet

This scalar object will be supported read-only.

5.6.3 sonetPathCurrentWidth

This scalar object will be supported read-only.

5.7 Paradyne SONET-EXT MIB (pdn_SonetExt.mib).

Table 1-35. SONET-EXT MIB

Table	Comments
devSonetConfigEntry_t	Supported as is

5.7.1 devSonetXmitClkSrc.

This MIB object is used to configure the clock source to be used for the Sonet interface in the transmit direction. **The default value is loopTiming (1)**.

5.7.2 devSonetStatusLastChange.

This MIB object corresponds to the sysUpTime since this Sonet interface entered the current line status.

5.7.3 devSonetStatusChangeTrapEnable.

This MIB object is used to enable and disable the generation of the Sonet Status change trap. **By default, this object is enabled.**

5.8 SHDSL MIB (RFC 3276)

Table 1-36. SHDSL MIB

Table	Comments
hdsl2ShdslSpanConfEntry	Supported as is
hdsl2ShdslSpanStatusEntry	Supported as is
hdsl2ShdslInventoryEntry	Supported as is
hdsl2ShdslEndpointConfEntry	Supported as is
hdsl2ShdslEndpointCurrEntry	Supported as is
hdsl2Shdsl15MinIntervalEntry	Supported as is
hdsl2Shdsl1DayIntervalEntry	Supported as is
hdsl2ShdslEndpointMaintEntry	Supported as is
hdsl2ShdslUnitMainEntry	Supported as is
hdsl2ShdslSpanConfProfileEntry	Supported as is
hdsl2ShdslEndpointAlarmConfProfileEntry	Supported as is

5.9 Spectrum Management (pdn_spectrummgmt.mib)

This MIB is a common MIB supported for both ReachDSL and SHDSL ports.

Table 1-37. Spectrum Management

Table	Comments
newSpectrumMgmtGeneralConfigEntry	Supported as is
newSpectrumMgmtConfEntry	Supported as is
newSpectrumMgmtLineInfoEntry	Supported as is

5.9.1 newSpectrumMgmtGeneralConfigEntry

This table is indexed by the entPhysicalIndex and ifType of the ports contained in the entity module (DSL port card). An entry in this table corresponds to the general spectrum management configuration objects of a DSL port card. Such configuration parameters control all DSL ports of a specific technology residents in that card. There will an entry for each card whose ports types are either D_ifType_shdsl or D_ifType_reachDSL.

6. Link Layer

6.1 ATM (AtoM) MIB (RFC 2515)

The ATM MIB provides objects for management of the ATM Cell and AAL5 interfaces. Although primarily associated with mib-2 37, RFC 2515 provides 9 groups for support of ATM. The following table indicates the support for these groups. The related MIB objects are supported in the DSLATM, ATMUP and AAL5 interfaces as they apply per RFC 2515. Please refer to Table 1-6, GranDSLAM 3.0 Interface List for the exact number of ATM interfaces.

Table 1-38. ATM MIB

Table	Comments
atmInterfaceConfEntry	Supported as is
atmInterfaceDs3PlcpEntry	Not supported
atmInterfaceTCEntry	Supported as is
atmTrafficDescrParamEntry	Supported as is
atmVplEntry	VP Switching is not supported
atmVclEntry	Supported as is
atmVpCrossConnectEntry	VP Switching is not supported
atmVcCrossConnectEntry	Supported as is
aal5VccEntry	Not supported

6.1.1 ATM Interface Configuration Parameter Group (RFC 2515)

This table contains information on ATM cell layer configuration of local ATM interfaces on an ATM device in addition to the information on such interfaces contained in the ifTable. This table is supported for each ATM interface: Backplane, and Management Processor. The table, atmInterfaceConfEntry, contains the following objects:

6.1.1.1 “atmInterfaceMaxVpcs” Object (atmInterfaceConfEntry 1)

This object contains the maximum number of VPCs supported at this ATM interface. The only valid values for this object are:

Table 1-39. atmInterfaceMaxVpcs

Interface	atmInterfaceMaxVpcs
DSLATM	0
ATMUP	0
AAL5	0

6.1.1.2 "atmInterfaceMaxVccs" Object (atmInterfaceConfEntry 2)

This object contains the maximum number of VCCs supported at this ATM interface. The only valid values for this object are:

Table 1-40. atmInterfaceMaxVccs

Interface	atmInterfaceMaxVccs
DSLATM	4
ATMUP	8192
AAL5	10

6.1.1.3 "atmInterfaceConfVpcs" Object (atmInterfaceConfEntry 3)

This object contains the number of VPCs configured for use at this ATM interface.

Table 1-41. atmInterfaceConfVpcs

Interface	atmInterfaceConfVpcs
DSLATM	0
ATMUP	0
AAL5	0

6.1.1.4 "atmInterfaceConfVccs" Object (atmInterfaceConfEntry 4)

This object contains the number of VCCs configured for use at this ATM interface.

Table 1-42. atmInterfaceConfVccs

Interface	atmInterfaceConfVccs
DSLATM	Actual number of VCCs
ATMUP	Actual number of VCCs
AAL5	Actual number of VCCs

6.1.1.5 "atmInterfaceMaxActiveVpiBits" Object (atmInterfaceConfEntry 5)

This object contains the maximum number of active VPI bits configured for use at this ATM interface.

Table 1-43. atmInterfaceMaxActiveVpiBits

Interface	atminterfaceMaxActiveVpiBits
DSLATM	1
ATMUP	2
AAL5	6

6.1.1.6 "atmInterfaceMaxActiveVciBits" Object (atmInterfaceConfEntry 6)

This object contains the maximum number of active VCI bits configured for use at this ATM interface.

Table 1-44. atmInterfaceMaxActiveVciBits

Interface	atmInterfaceMaxActiveVciBits
DSLATM	6
ATMUP	10
AAL5	6

6.1.1.7 "atmInterfaceIlmiVpi" Object (atmInterfaceConfEntry 7)

6.1.1.8 This object contains the VPI value of the VCC supporting the ILMI on the ATM Interface. **"atmInterfaceIlmiVci" Object (atmInterfaceConfEntry 8)**

Table 1-45. atmInterfaceIlmiVpi

Interface	atmInterfaceIlmiVpi	Access
DSLATM	0	Read-Only
ATMUP	0	Read-Only
AAL5	0	Read-Only

This object contains the VCI value of the VCC supporting the ILMI on the ATM Interface. .

Table 1-46. atmInterfaceIlmiVci

Interface	atmInterfaceIlmiVci	Access
DSLATM	0	Read-Only
ATMUP	16	Read-Only
AAL5	0	Read-Only

6.1.1.9 "atmInterfaceMyNeighborIpAddress" Object (atmInterfaceConfEntry 11)

In Hybrid Management Model, the value of this object contains 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.

Implementation Note: for those ATM interfaces over xDSL interfaces, the value of this object is obtained from NMS station when the endpoint IP address is assigned for the purpose of the persistent management of the endpoint.

Write access to this object is not supported

Table 1-47. atmInterfaceMyNeighborIpAddress

Interface	atmInterfaceMyNeighborIpAddress	Access
DSLATM	0.0.0.0	Read-Only
ATMUP	0.0.0.0	Read-Only
AAL5	0.0.0.0	Read-Only

6.1.1.10 "atmInterfaceMyNeighborIfName" Object (atmInterfaceConfEntry 12)

In Hybrid Management Model, the value of this object contains 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.

Write access to this object is not supported

Table 1-48. atmInterfaceMyNeighborIfName

Interface	atmInterfaceMyNeighborIfName	Access
DSLATM	zero length string	Read-Only
ATMUP	zero length string	Read-Only
AAL5	zero length string	Read-Only

6.1.1.11 "atmInterfaceCurrentMaxVpiBits" Object (atmInterfaceConfEntry 13)

The maximum number of VPI Bits that may currently be used at this ATM interface.

Table 1-49. atmInterfaceCurrentMaxVpiBits

Interface	atmInterfaceCurrentMaxVpiBits
DSLATM	1
ATMUP	4
AAL5	6

6.1.1.12 "atmInterfaceCurrentMaxVciBits" Object (atmInterfaceConfEntry 14)

The maximum number of VCI Bits that may currently be used at this ATM interface.

Table 1-50. atmInterfaceCurrentMaxVciBits

Interface	atmInterfaceCurrentMaxVpiBits
DSLATM	6
ATMUP	10
AAL5	6

6.1.1.13 "atmInterfaceSubscrAddress" Object (atmInterfaceConfEntry 15)

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.

Table 1-51. atmInterfaceSubscrAddress

Interface	atmInterfaceSubscrAddress
DSLATM	zero length string
ATMUP	zero length string
AAL5	zero length string

6.1.2 TC Sublayer Group (RFC 2515)

This table, atmInterfaceTCEntry, provides performance statistics of the TC sublayer of local ATM interfaces on a managed ATM device. **This table is not supported**.

6.1.2.1 "atmInterfaceOCDEvents" Object (atmInterfaceTCEntry 1)

Table 1-52. atmInterfaceOCDEvents

Interface	atmInterfaceOCDEvents
DSLATM	Not supported
ATMUP	Not supported

6.1.2.2 "atmInterfaceTCAlarmState" Object (atmInterfaceTCEntry 2)

Table 1-53. atmInterfaceTCAlarmState

Interface	atmInterfaceTCAlarmState
DSLATM	NoAlarm(1) LcdFailure(2)
ATMUP	NoAlarm(1) LcdFailure(2)

6.2 ATM Traffic Descriptor Group (RFC 2515)

This 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 will be 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.

Table 1-54. Traffic Descriptor Types

atmNoClpNoScr
atmNoClpScr
atmClpNoTaggingScr
atmClpTaggingScr
atmClpNoTaggingMcr
atmClpTransparentNoScr
atmClpTransparentScr
atmNoClpTagingNoScr
atmNoClpNoScrCdvt
atmnoClpScrCdvt
atmClpNoTaggingScrCdvt
atmClpTaggingScrCdvt

6.2.0.1 “atmServiceCategory” Object (atmTrafficDescrParamEntry 10)

The only valid value for this object is :

- UBR(6) -- unspecified bit rate.
- CBR(2) - constant bit rate
- nrt-VBR(4) - Variable bit rate - non real time
- rt-VBR(3) - Variable bit rate - real time

6.2.1 Virtual Channel Link (VCL) Configuration Group (RFC 2515)

This table contains configuration and state information of a bi-directional Virtual Channel Link (VCL) at an ATM interface. 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. ATM Interface Virtual Channel Link (VCL) Table, atmVclEntry is fully supported and contains the following objects:

6.2.1.1 "atmVclOperStatus" Object (atmVclEntry 4)

This object indicates the current operational status of the VCL. Possible values are:

atmVclOperStatus =
- up(1) - VCL is currently Operational.
- down(2) - VCL is currently Not Operational.
- unknown(3) - The state of the VCL can't be determined. For example-all VCLs for a given interface will be created automatically once the box comes up initially. These VCLs will be given an OperStatus of unknown.

6.2.1.2 "atmVclLastChange" Object (atmVclEntry 5)

This object contains the value of the MIB II's sysUpTime object at the time the VCL 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.

6.2.1.3 "atmVccAalType" Object (atmVclEntry 8)

This object identifies the type of AAL used with the VCC. The only valid values for this object is:

- aal5(3)- AAL5
- unknown(5) - The unknown type indicates that the AAL type cannot be determined.

6.2.1.4 "atmVccAal5CpcsTransmitSduSize" Object (atmVclEntry 9)

Not supported.

This object contains the maximum AAL5 CPCS (Common Part Convergence Sublayer) SDU size in octets that is supported on the transmit direction of this VCC. :An instance of this object only exists when the local VCL end-point is also the VCC end-point, and AAL5 is in use

atmVccAal5CpcsTransmitSduSize = 1500 GranDSLAM 3.0 supports a maximum user data size of 4096.

6.2.1.5 "atmVclAal5CpcsReceiveSduSize" Object (atmVclEntry 10)

Not supported.

This object contains the maximum AAL5 CPCS (Common Part Convergence Sublayer) SDU size in octets that is supported on the receive direction of this VCC. For 4200 stackable DSLAM, possible values are:

atmVccAal5CpcsReceiveSduSize = 4096 - GranDSLAM 3.0 supports a maximum user data size of 4096.

6.2.1.6 "atmVclEncapsType" Object (atmVclEntry 11)

Not supported.

This object identifies the type of data encapsulation used over the AAL5 SSCS (Service Specific Convergence Sublayer) layer. The only valid values for this object is:

atmVclEncapsType = unknown(10) - when the ATM cell interface is NOT the SAR.

6.2.2 Virtual Channel Cross Connect Group (RFC 2515)

This table contains configuration and state information of point-to-point, point-to-multipoint or multipoint-to-multipoint VC cross-connects for PVCs. 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. The ATM Virtual Channel (VC) Cross Connect Table, atmVcCrossConnectEntry table contains the following objects:

6.2.3 "atmVcCrossConnectL2HOperStatus" Object (atmVcCrossConnectEntry 9)

The value of this object identifies the current operational status of the VC cross-connect in one direction; (that is, from the low to high direction). The up and down states indicate that this ATM VC cross-connect from low to high direction is operational or not operational respectively. The unknown state indicates that the state of it cannot be determined. The only valid values for this object are:

- up(1)
- down(2)
- unknown (3) (NOT_SUPPORTED)

6.2.4 "atmVcCrossConnectH2LOperStatus" Object (atmVcCrossConnectEntry 10)

The value of this object identifies the current operational status of the VC cross-connect in one direction; (that is, from the high to low direction). The up and down states indicate that this ATM VC cross-connect from high to low direction is operational or not operational respectively. The unknown state indicates that the state of it cannot be determined. The only valid values for this object are:

- up(1)
- down(2)
- unknown (3) (NOT_SUPPORTED)

6.2.5 "atmVcCrossConnectL2HLastChange" Object (atmVcCrossConnectEntry 11)

The value of MIB II's sysUpTime object at the time this VC cross-connect entered its current operational state in low to high direction. If the current state was entered prior to the last re-initialization of the agent, then this object contains a zero value.

6.2.6 "atmVcCrossConnectH2LLastChange" Object (atmVcCrossConnectEntry 12)

The value of MIB II's sysUpTime object at the time this VC cross-connect entered its current operational state in low to high direction. If the current state was entered prior to the last re-initialization of the agent, then this object contains a zero value.

6.2.7 "atmVcCrossConnectRowStatus" Object (atmVcCrossConnectEntry 13)

The status of this entry in the atmVcCrossConnectTable. This object is used to create a new cross-connect row 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'. This object cannot be set to 'active' unless the following columnar object exists in this row: atmVcCrossConnectAdminStatus. To turn on a VC cross-connect, the atmVcCrossConnectAdminStatus is set to 'up'. The only valid values for this object are:

- active
- createAndGo
- destroy

6.3 atmfM4MIB- SNMP M4 Network Element View MIB - [atmfM4MIB]

The ATM Forum MIB - (atmfM4MIB.MIB). As defined in the ATM Forum SNMP M4 Network Element View MIB (AF-NM-0095.001) document. **The table below shows the object of this MIB that are supported.** For the GranDSLAM 3.0 product VP switching is not supported.

Table 1-55. Object Supported in ATM Forum MIB

Table	Object	Type	
atmfM4TcAdapterEntry	Note: Only 'false (2)' is supported for atmfM4TcACellScrambling		Yes
	atmfM4TcCellScrambling(1)	TruthValue	Read-Only
	atmfM4TcAlarmSeverityIndex(2)	Integer32	No
atmfM4VclEntry	Supported as is		
atmfM4ATMLayerEntry	Supported (read-only)		
atmfM4VclEntry	Supported as is		
atmfM4VcXConnEntry	Supported as is		
atmfM4VcNextVciEntry	Supported as is		
atmfM4CellProtoCurrEntry	These tables are read-only by definition and are supported as is.		
atmfM4CellProtoHistEntry			
atmfM4CellProtoErrorEntry			
atmfM4TcProtoCurrEntry			
atmfM4TcProtoHistEntry			
atmfM4VcUpcNpcCurrEntry			
atmfM4VcUpcNpcHistEntry			
atmfM4VcTestEntry	VC testing is supported		

6.3.1 TC Adapter Layer Table (atmfM4TcAdapterEntry)

Columns conceptually added to the interface table entry for an ATM interface to model the TC Adapter. The row of the interface table modeling this object should also include columns for the physical path TP and the ATM interface table defined in RFC 2515. The interface table entry for the TC Adapter fields of an ATM interface are set up without management system control (or else that setup is outside the scope of M4).

6.3.1.1 "atmfM4TcAlarmSeverityIndex"Object (atmfM4TcAdapterEntry 2)

Specifies the index of the entry in the communications alarm severity profile table that should be used. The only valid value of this object is 0.

6.3.1.2 ATM Cell Layer Table (atmfM4ATMLayerEntry)

These are the ATM Forum M4 interface Configuration table extensions for the ATM cell layer. The default configuration of an entry in this table is used whenever the entries for the physical path TP, etc., are created. The management system configures a UNI, B-ICI, or B-ISSI on the interface by first modifying the atmfM4IfType column (this can be done in the same SNMP Set-request that sets up the other necessary variables.

6.3.1.3 "atmfM4IfType" Object (atmfM4ATMLayerEntry 1)

Always uni(1).

6.3.1.4 "atmfM4IfSubscriberAddress" Object (atmfM4ATMLayerEntry 3)

Returns zero length string. Used for SVCs.

6.3.1.5 "atmfM4IfPreferredCarrier" Object (atmfM4ATMLayerEntry 4)

Returns zero length string. Used for SVCs.

6.3.1.6 "atmfM4IfFarEndCarrierNetwork" Object (atmfM4ATMLayerEntry 5)

Returns zero length string. Used for SVCs in B-ICI interface.

6.3.2 VCL Termination Point Configuration Table (atmfM4VclEntry)

6.3.2.1 "atmfM4VclSegEndPt" Object (atmfM4VclEntry 1)

Specifies whether the VCL termination point is a segment end-point. The only valid value for this object is:

- false (2)

6.3.3 VC Cross-Connect Table (atmfM4VcXConnEntry)

This table augments the VC Cross-Connect configuration table in RFC 2515.

6.3.3.1 "atmfM4VcXConnRecover" Object (atmfM4VcXConnEntry1)

The only valid value for this object is: true.

6.3.4 ATM Cell Protocol Monitoring Current Data (atmfM4CellProtoCurrEntry)

Each ATM interface automatically has an entry in this table associated with it This table maintains per-interface statistics for the fifteen-minute interval currently being collected. Supported by unknown cell process.

6.3.4.1 "atmfM4CellProtoCurrInOAMCells" Object (ATMfM4CellProtoCurrEntry 3)

Count of OAM Cells on this interface.

6.3.5 ATM Cell Protocol Monitoring History Data (atmfM4CellProtoHistEntry)

This table maintains per-interface statistics for previous fifteen-minute intervals.

6.3.5.1 "atmfM4CellProtoHistInOAMCells"Object (atmfM4CellProtoHistEntry 6)

Count of OAM Cells on this interface.

6.4 Paradyne Extensions to the ATM Interface Config Table and ATM Forum M4.

The following MIBs are intended to augment ATM Interface Config Table and ATM Forum M4 MIBs. pdn_ATMM4Ext.mib.

Table 1-56. Paradyne Extensions to ATM MIBs (1 of 2)

Table	Object	Type	Sup-ported
pdnATMIfConfExtEntry pdn_ATMM4Ext.mib			Yes
	pdnATMIfConfExtVbrRtBandwidthUtil	Integer32	Read-Only
	pdnATMIfConfExtVbrNrtBandwidthUtil	Integer32	Read-Only
	pdnATMIfConfExtHecErrorThreshold	Integer32	No
	pdnATMIfConfExtUnknownCellThreshold	Integer32	No
	pdnATMIfConfExtOcdEventThreshold	Integer32	No
	pdnATMIfConfExtBandwidthUtilCbrReserved	Integer32	No
	pdnATMIfConfExtBandwidthUtilCbrAssigned	Integer32	No
	pdnATMIfConfExtBandwidthUtilVbrRtReserved	Integer32	No
	pdnATMIfConfExtBandwidthUtilVbrRtAssigned	Integer32	No
	pdnATMIfConfExtBandwidthUtilVbrNrtReserved	Integer32	No
	pdnATMIfConfExtBandwidthUtilVbrNrtAssigned	Integer32	No
	pdnATMIfConfExtBandwidthUtilUbrReserved	Integer32	No
	pdnATMIfConfExtBandwidthUtilUbrAssigned	Integer32	No
pdnATMTrafficDescrParamExtEntry pdn_ATMM4Ext.mib	Supported as is		
pdnATMfM4TcProtoCurrExtEntry pdn_ATMM4Ext.mib	Supported as is		
pdnATMfM4TcProtoHistExtEntry pdn_ATMM4Ext.mib			
pdnATMfM4VcLoopbackTable pdn_ATMM4Ext.mib			
pdnATMfM4Vc1CellLoopTable pdn_ATMM4Ext.mib			Yes
pdnATMfM4LoopbackLocationCode pdn_ATMM4Ext.mib			Yes
	pdnATMfM4LoopbackLocationCode	OctetString	Yes

Table 1-56. Paradyne Extensions to ATM MIBs (2 of 2)

Table	Object	Type	Sup-ported
pdnATMVclCurrEntry			Yes
(pdn_ATMStats.mib)	pdnATMVclCurrElapsedTime	Gauge32	Read-Only
	pdnATMVclCurrTotalCellIns	Gauge32	Read-Only
	pdnATMVclCurrTotalInDiscards	Gauge32	n
	pdnATMVclCurrTotalCellOuts	Gauge32	Read-Only
	pdnATMVclCurrTotalOutDiscards	Gauge32	Read-Only
pdnATMVclHistEntry	pdnATMVclHistIndex	INTEGER	Read-Only
(pdn_ATMStats.mib)	pdnATMVclHistElapsedTime	TimeInterval	Read-Only
	pdnATMVclHistTotalCellIns	Gauge32	Read-Only
	pdnATMVclHistTotalInDiscards	Gauge32	n
	pdnATMVclHistTotalCellOuts	Gauge32	Read-Only
	pdnATMVclHistTotalOutDiscards	Gauge32	Read-Only

6.5 ATM Loopback

pdnATMfM4VcLoopbackTestTable in pdn_atmM4Ext.mib is supported to provide detailed test results on ATM F5 loopback test. This tabular object augments atmfM4VcTestTable (see Section 3.15).

Table 1-57. ATM Loopback

Table	Comments
pdnATMfM4VcLoopbackTestEntry	Supported as is

In order to distinguish a single-cell loopback test from a multi-cell loopback test, In addition to ATM Forum M4 Test Types atmfM4VcTestOAMLoopbackSeg and atmfM4VcTestOAMLoopbackE2E, which indicate a single-cell loopback test, two new test commands pdnATMfM4TestOAMLoopbackSegMultiCell and pdnATMfM4VcTestOAMloopbackE2EMultiCell are defined to stop but not dealloc the ATM F5 loopback test resource (such as result).

In order to initiate an ATM F5 Segment loopback test, an NMS must first obtain ‘ownership’ of the entry in the atmfM4VcTestTable for the VCC or VCL to be tested. This is accomplished with atmfM4VcTestId and atmfM4VcTestStatus object as follows:

```
try_again:
    get (atmfM4VcTestId, atmfM4VcTestStatus)
    while (atmfM4TestStatus != notInUse)
/* Loop while a test is running or some other
     * manager is configuring a test.
     */
    short delay
```

```
    get (atmfM4TestId, atmfM4TestStatus)
}
/*
 * Is not being used right now -- let's compete
 * to see who gets it.
 */
lock_value = atmfM4TestId

if ( set(atmfM4TestId = lock_value, atmfM4TestStatus = inUse,
        atmfM4TestOwner = 'my-IP-address') == FAILURE)
    /*
     * Another manager got the atmfM4TestEntry -- go
     * try again
     */
    goto try_again;
                /*
                    * I have the lock
        */
set up any test parameters.

/*
 * This starts the test
 */
set(atmfM4TestType = test_to_run);
            set(atmfM4TestCode = pdnATMfM4VcLoopbackTestTable);

when test is stopped by this NMS station, agent sets atmfM4TestResult
    agent also sets atmfM4TestStatus = 'notInUse'

retrieve any additional test results, and atmfM4TestId
if (atmfM4TestId == lock_value+1) results are valid
```

Once ownership is obtained, an NMS could initiate a test by setting atmfM4VcTestType to pdnATMfM4VcTestOAMLoopbackSegMultiCell to which a loopback location code that uniquely identifies an intermediate point responsible for looping back the cell is appended for ATM OAM F5 Segment loopback test, or pdnATMfM4VcTestOAMLoopbackE2EMultiCell for ATM OAM F5 End to End loopback test. If the loopback location code is absent from the pdnATMfM4VcTestOAMLoopbackSegMultiCell, the loopback is performed at the end of the segment or connection. The Ubiquity ATM unit supports next-hop segment ATM loopback, which means that the Ubiquity ATM unit ignores the loopback location code if the NMS application provides one.

Once a test is started, the SNMP agent will set the value of atmfM4VcTestCode to 1.3.6.1.4.1.1795.2.24.2.6.11.6.1.4 (pdnATMfM4VcLoopbackTestTable) indicating the test statistics are in pdnATMfM4VcLoopbackTestTable. The NMS can retrieve the test results by querying the objects in this table.

The NMS station, the owner of the ongoing OAM F5 loopback test, must explicitly stop this loopback test by setting atmfM4VcTestType to NoTest (0.0) when the NMS no longer needs this test The test result might be available depending upon whether there is enough room in the system.

At any time, an NMS station can set atmfM4VcTestType to “0.0” to abort the ongoing test owned by this NMS.

6.6 IMA MIB (AF-PHY-0086.001)

Table 1-58. IMA MIB

Table	Object
imaGroupNumber	Supported as is (a max of 8 IMA groups)
imaGroupEntry	Supported as is
imaGroupMappingEntry	Supported as is
ImaLinkEntry	Supported as is
ImaAlarmStatus	Supported as is
imaAlarmType	Supported as is
imaGroupCurrentEntry	Supported as is
ImaGroupIntervalEntry	Supported as is
ImaGroupTotalEntry	Supported as is
imaLinkCurrentEntry	Supported as is
imaLinkTotalEntry	Supported as is
imaLinkIntervalEntry	Supported as is

7. Configuration Management

7.1 Entity MIB Version 2 - (RFC 2737)

7.1.1 MIB Structure.

The Entity MIB (version 2) contains five groups of MIB objects:

- entityPhysical group

Describes the physical entities managed by a single agent.

- entityLogical group

Describes the logical entities managed by a single agent.

- 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.

- entityNotifications group

Contains status indication notifications.

7.1.2 Entity MIB (version 2) - Objects supported.

This section describes more in detail how the different MIB objects will be suppported. It also describes the different classes of physical entities, their relationships and the ifIndex scheme to reference each physical entity in the system.

7.1.2.1 entityPhysical Group.

The entPhysicalTable { entityPhysical 1} described below shows the attributes supported for each the physical entities in the ATM stackable product.

Table 1-59. entPhysicalEntry {entPhysicalTable 1} (1 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
entPhysicalIndex	{entPhysicalEntry 1}	PhysicalIndex	not-accessible	current	Yes
entPhysicalDesc	{entPhysicalEntry 2}	SnmpAdminString	read-only	current	Yes
entPhysicalVendorType	{entPhysicalEntry 3}	AutonomousType	read-only	current	Yes
entPhysicalContainedIn	{entPhysicalEntry 4}	INTEGER	read-only	current	Yes
entPhysicalClass	{entPhysicalEntry 5}	PhysicalClass	read-only	current	Yes
entPhysicalParentRelPos	{entPhysicalEntry 6}	INTEGER	read-only	current	Yes
entPhysicalName	{entPhysicalEntry 7}	SnmpAdminString	read-only	current	Yes
entPhysicalHardwareRev	{entPhysicalEntry 8}	SnmpAdminString	read-only	current	Yes
entPhysicalFirmwareRev	{entPhysicalEntry 9}	SnmpAdminString	read-only	current	Yes
entPhysicalSoftwareRev	{entPhysicalEntry 10}	SnmpAdminString	read-only	current	Yes
entPhysicalSerialNum	{entPhysicalEntry 11}	SnmpAdminString	read-write	current	Yes
entPhysicalMfgName	{entPhysicalEntry 12}	SnmpAdminString	read-only	current	Yes

Table 1-59. entPhysicalEntry {entPhysicalTable 1} (2 of 2)

Object	OID	Syntax	Access	Status	Sup-ported
entPhysicalModelName	{entPhysicalEntry 13}	SnmpAdminString	read-only	current	Yes
entPhysicalAlias	{entPhysicalEntry 14}	SnmpAdminString	read-write	current	Yes
entPhysicalAssetID	{entPhysicalEntry 15}	SnmpAdminString	read-write	current	Yes
entPhysicalIsFRU	{entPhysicalEntry 16}	Truth Value	read-only	current	Yes

7.1.2.1.1 entPhysicalIndex

The index for this entry.

Valid Object Value(s):

The value of this object is based on the ifIndex in the Interfaces Group. Refer to *Evolution of the Interfaces Group of MIB-II (RFC 2863)* on page 1-18 for further details of these indexes. The following are the values for the entPhysicalIndex object based on that scheme.

Table 1-60. entPhysicalIndex

Entity	R	R	C	C	S	S	L	P	P	P	Comments
Chassis	0	0	0	1	0	0	0	0	0	0	Chassis
Container	0	0	0	1	Y	Y	0	0	0	0	Slot ’yy’, yy=[1..20]
Module	0	0	0	1	Y	Y	0	0	0	1	Module (SCPs and ATM portcards) in slot ’yy’, yy=[1..20]
Splitter card	0	0	0	1	Y	Y	0	0	0	3	Splitter in slot ’yy’, yy=[1..20]
DSL Port	0	0	0	1	Y	Y	1	0	Z	Z	Dsl user ports, yy=[1..20], zz=[1..48]
t1e1 port	0	0	0	1	0	0	1	9	0	Z	T1/E1 ports, z=[1..8]
OC3 port	0	0	0	1	0	0	1	9	0	Z	OC3 ports, z=[1..2]
Console port	0	0	0	1	0	0	1	8	0	1	Console port
Ethernet Mgmt port	0	0	0	1	0	0	1	8	0	2	Ethernet port
Temp Sensor	0	0	0	1	0	0	0	1	0	Z	Temperature Sensor in chassis ‘z’, z=[1..3]
Fan	0	0	0	1	0	0	0	1	1	Z	Fan ‘z’in Chassis, z=[1]
Management Plane PLD	0	0	0	1	Y	Y	0	1	2	1	Management Plane PLD in slot yy, yy=[1..20]
Power Supply	0	0	0	1	0	0	0	1	3	Z	Power Supply ‘z’in Chassis, z=[1..2]

7.1.2.1.2 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 distint value for each version or model of the physical entity.

Valid Object Value(s):

Table 1-61. entPhysicalDescr

Entity	Description
Chassis	sysDescr as described in the MIB-II's System Group.
Container	"Paradyne DSLAM Slot x", Where x=1..20
Splitter card	Paradyne DSLAM ; Splitter Module; Unit n
DSL Port	ifDescr from Interfaces group (Section 4)
T1/E1 port	ifDescr from Interfaces group (Section 4)
OC3 port	ifDescr from Interfaces group (Section 4)
Console port	Console port
Ethernet Mgmt port	ifDescr from Interfaces group (Section 4)
Temp Sensor	Temperature Sensor #x[c]; Unit n
Fan	Fan #y[d].; Unit n
Power Supply	Power Supply #z[e].; Unit n
Management Plane PLD	Management Plane PLD; Unit n

a. Refers to the sysDescr in the MIB-II's System Group for clarification on Company Name.

b. n is the unit number that ranges between 1..8

c. x is the sensor number that ranges between 1..3

d. y is the fan number that ranges between 1..1

e. z is the power supply number that ranges between 1..2

7.1.2.1.3 entPhysicalVendorType

An indication of the vendor-specific hardware type of the physical entity.

Valid Object Value(s):

The following is the OID tree for the different entities supported.

Company Enterprise.1.14.9.14.5 (sme-chassis)

.1 sme-chassis-8820

.2 sme-chassis-8620

Company Enterprise.1.14.9.14.7 (sme-container)

Company Enterprise.1.14.9.14.1 (sme-scp-cards)

.1 sme-scp-8412-card (SCP w/OC3/STM1 MM ATM Uplink)

.2 sme-scp-8413-card (SCP w/OC3/STM1 SM int ATM Uplink)

.3 sme-scp-8414-card (SCP w/OC3/STM1 SM long ATM Uplink)

.4 sme-scp-8416-card (SCP w/8 port T1 IMA uplink)

.5 sme-scp-8417-card (SCP w/8 port E1 IMA uplink 75 ohm)

.6 sme-scp-8418-card (SCP w/8 port E1 IMA uplink 120 ohm)

Company Enterprise.1.14.9.14.4 (sme-portcards)

.1 sme-8965-card

.2 sme-8955-card

.3 sme-8985-card

Company Enterprise.1.14.9.14.6 (sme-components)

.1 sme-comp-fan

.2 sme-comp-pld

.3 sme-comp-sensor

.4 sme-comp-powersupply

The following table show the entPhysicalVendorType based on the tree OID above.

Table 1-63. entPhysicalVendorType (1 of 2)

Entity	entPhysicalVendorType
Chassis	{ *Company Enterprise*.1.14.9.14.5.1} sme-chassis-8820 { *Company Enterprise*.1.14.9.14.5.2} sme-chassis-8620
Container	{ *Company Enterprise*.1.14.9.14.7} sme-container
Module	{ *Company Enterprise*.1.14.9.14.1.1} sme-scp-8412-card { *Company Enterprise*.1.14.9.14.1.2} sme-scp-8413-card { *Company Enterprise*.1.14.9.14.1.3} sme-scp-8414-card { *Company Enterprise*.1.14.9.14.1.4} sme-scp-8416-card { *Company Enterprise*.1.14.9.14.1.5} sme-scp-8417-card { *Company Enterprise*.1.14.9.14.1.6} sme-scp-8418-card { *Company Enterprise*.1.14.9.14.4.1} sme-8965-card { *Company Enterprise*.1.14.9.14.4.1} sme-8955-card { *Company Enterprise*.1.14.9.14.4.1} sme-8985-card
Splitter card	{ *Company Enterprise*.1.14.9.14.2.2} sme-int-splitter
DSL Port	{ *Company Enterprise*.1.14.9.14.3.1 } sme-reachDsl-port { *Company Enterprise*.1.14.9.14.3.2 } sme-adsl-a-port { *Company Enterprise*.1.14.9.14.3.3 } sme-adsl-b-port { *Company Enterprise*.1.14.9.14.3.4 } sme-shdsl-port

Table 1-63. entPhysicalVendorType (2 of 2)

Entity	entPhysicalVendorType
T1/E1 port	{ *Company Enterprise*.1.14.9.14.3.5 } sme-t1e1-port
OC3 port	{ *Company Enterprise*.1.14.9.14.3.6 } sme-oc3-port
Console port	{ *Company Enterprise*.1.14.9.14.3.4 } sme-rs232-port
Ethernet Mgmt port	{ *Company Enterprise*.1.14.9.14.3.4 } sme-eth-port
Temp Sensor	{ *Company Enterprise*.1.14.9.14.6.3}sme-comp-sensor
Fan	{ *Company Enterprise*.1.14.9.14.6.1}sme-comp-fan
Power Supply	{ *Company Enterprise*.1.14.9.14.6.4}sme-comp-powersupply
Management Plane PLD	{ *Company Enterprise*.1.14.9.14.6.2}sme-comp-pld

7.1.2.1.4 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..

Valid Object Value(s):

The highest entity in the hierarchy is the stack entity which is not contained in any other entity. The following table show the entPhysicalContainedIn relationship between the entities.

Table 1-64. entPhysicalContainedIn

Entity	entPhysicalContainIn
Chassis	0
Container	entPhysicalIndex for the corresponding Chassis entity
Module	entPhysicalIndex for the corresponding Container entity
Splitter card	entPhysicalIndex for the corresponding Container entity
DSL Port	entPhysicalIndex for the corresponding module entity
T1/E1 port	entPhysicalIndex for the corresponding SCP entity card
OC3 port	entPhysicalIndex for the corresponding SCP entity card
Console port	entPhysicalIndex for the corresponding SCP entity card
Ethernet Mgmt port	entPhysicalIndex for the corresponding SCP entity card
Temp Sensor	entPhysicalIndex for the corresponding card entity
Fan	entPhysicalIndex for the corresponding Chassis entity
PowerSupply	entPhysicalIndex for the corresponding Chassis entity
Management Plane PLD	entPhysicalIndex for the corresponding card entity

7.1.2.1.5 entPhysicalClass

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

Valid Object Value(s):

The following table shows the different classes identified for the ATM stackable DSLAM's entities.

Table 1-65. entPhysicalClass

Entity	entPhysicalClass
Chassis	chassis (3)
Container	container (5)
Module	module (9)
Splitter card	module (9)
DSL Port	port (10)
T1/E1 port	port (10)
OC3 port	port (10)
Console port	port (10)
Ethernet Mgmt port	port (10)
Temp Sensor	sensor (8)
Fan	fan (7)
PowerSupply	powersupply (6)
Management Plane PLD	other (1)

7.1.2.1.6 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..

Valid Object Value(s):

The following table shows the entPhysicalParentRelPos relationship among the entities.

Table 1-66. entPhysicalParentRelPos (1 of 2)

Entity	entPhysicalParentRelPos
Chassis	-1
Container	Position of this container in the chassis [1..20]
Module	Position of this module in the slot container. Should be equal to 1
Splitter card	3
DSL Port	A value between [1..24]

Table 1-66. entPhysicalParentRelPos (2 of 2)

Entity	entPhysicalParentRelPos
T1/E1 port	A value between [3..11]
OC3 port	3
Console port	1
Ethernet Mgmt port	2
Temp Sensor	A value between [1..3]
Fan	A value between [1..1]
PowerSupply	A value between [1..2]
Management Plane PLD	1

7.1.2.1.7 entPhysicalName

The textual name of the physical entity.

Valid Object Value(s):

Table 1-67. entPhysicalName

Entity	entPhysicalName
Chassis	“Chassis 8620”, "Chassis 8820"
Container	“Slot"
Module	“SCP", "Shdsl", "ReachDSL", "ADSL"
Splitter card	“SplitterCard”
DSL Port	ifName from the ifXTable
T1/E1 port	ifName from the ifXTable
OC3 port	ifName from the ifXTable
Console port	“Console”
Ethernet Mgmt port	ifName from the ifXTable
Temp Sensor	“T_Sensor n”
Fan	“Fan n”
PowerSupply	“PowerSupply n”
Management Plane PLD	“MP_PLD”

7.1.2.1.8 entPhysicalHardwareRev

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

Valid Object Value(s):

The following table shows the entPhysicalHardwareRev defined for each entity.

Table 1-68. entPhysicalHardwareRev

Entity	entPhysicalHardwareRev
Chassis	zero-length-string
Container	zero-length-string
Module	String containing the Revision Number of the CCA
Splitter card	String containing the Revision Number of the CCA
DSL Port	zero-length-string
T1/E1 port	zero-length-string
OC3 port	zero-length-string
Console port	zero-length-string
Ethernet Mgmt port	zero-length-string
Temp Sensor	zero-length-string
Fan	zero-length-string
PowerSupply	zero-length-string
Management Plane PLD	zero-length-string

7.1.2.1.9 entPhysicalSoftwareRev:

The vendor-specific software revision string for the physical entity. 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. For the GranDSLAM 3.0 units, this object will always return a zero-length string.

7.1.2.1.10 entPhysicalFirmwareRev

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

Valid Object Value(s):

The following table shows the entPhysicalFirmwareRev assigned for the entities.

Table 1-69. entPhysicalFirmwareRev (1 of 2)

Entity	entPhysicalFirmwareRev
Chassis	zero-length-string
Container	zero-length-string
Module	A String with the currently Firmware running in this module.
Splitter card	zero-length-string
DSL Port	A String reflecting the DSP revisions
T1/E1 port	zero-length-string
OC3 port	zero-length-string

Table 1-69. entPhysicalFirmwareRev (2 of 2)

Entity	entPhysicalFirmwareRev
Console port	zero-length-string
Ethernet Mgmt port	zero-length-string
Temp Sensor	zero-length-string
Fan	zero-length-string
PowerSupply	zero-length-string
Management Plane PLD	A String with the currently running Software Revision Level.

7.1.2.1.11 entPhysicalSerialNum

The vendor-specific serial number string for the physical entity. Valid Object Value(s):

The following table shows the entPhysicalSerialNum identified for the entities supported.

Table 1-70. entPhysicalSerialNum

Entity	entPhysicalSerialNum
Chassis	zero-length-string
Container	zero-length-string
Module	A String with the Serial Number of the entity card.
Splitter card	zero-length-string
DSL Port	zero-length-string
T1/E1 port	zero-length-string
OC3 port	zero-length-string
Console port	zero-length-string
Ethernet Mgmt port	zero-length-string
Temp Sensor	zero-length-string
Fan	zero-length-string
PowerSupply	zero-length-string
Management Plane PLD	zero-length-string

7.1.2.1.12 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 following table shows the entPhysicalMfgName designated for the entities.

Table 1-71. entPhysicalMfgName

Entity	entPhysicalMfgName
Chassis	“Paradyne”
Container	"Paradyne"
Module	“Paradyne”
Splitter card	“Paradyne”
DSL Port	“A String reflecting the DSP Manufacturer”
T1/E1 port	“Paradyne”
OC3 port	“Paradyne”
Console port	“Paradyne”
Ethernet Mgmt port	“Paradyne”
Temp Sensor	“Paradyne”
Fan	“Paradyne”
Management Plane PLD	“Paradyne”
Led PLD	“Paradyne”
T1/E1 PLD	“Paradyne”

7.1.2.1.13 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):

The following table shows the entPhysicalModelName designated for the entities..

Table 1-72. entPhysicalModelName (1 of 2)

Entity	entPhysicalModelName
Chassis	zero-length-string
Container	zero-length-string
Module	A String describing the Model Number of the Module
Splitter card	A String describing the Model Number of the Splitter Card
DSL Port	zero-length-string
T1/E1 port	zero-length-string
OC3 port	zero-length-string
Console port	zero-length-string

Table 1-72. entPhysicalModelName (2 of 2)

Entity	entPhysicalModelName
Ethernet Mgmt port	zero-length-string
Temp Sensor	zero-length-string
Fan	zero-length-string
PowerSupply	zero-length-string
Management Plane PLD	zero-length-string

7.1.2.1.14 entPhysicalAlias

This object is an 'alias' name for the physical entity as specified by a network manager, and provides a non-volatilehandle' for the physical entity. On the first instantiation of an physical entity, the valueof 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 reinitializations/reboots of the network management system, including those which result in a change of the physical entity's entPhysicalIndex value.

Valid Object Value(s):

The following table shows the entPhysicalAlias assigned by default to the entities. This object is supported R/W.

Table 1-73. entPhysicalAlias

Entity	entPhysicalAlias
Chassis	zero-length-string
Container	zero-length-string
Module	zero-length-string
Splitter card	zero-length-string
DSL Port	zero-length-string
T1/E1 port	zero-length-string
OC3 port	zero-length-string
Console port	zero-length-string
Ethernet Mgmt port	zero-length-string
Temp Sensor	zero-length-string
Fan	zero-length-string
PowerSupply	zero-length-string
Management Plane PLD	zero-length-string

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

Valid Object Value(s):

The following table shows the entPhysicalAssetID assigned by default to the entities. NMS systems can modify these default values for some entities as indicated below.

Table 1-74. entPhysicalAssetID

Entity	entPhysicalAssetID
Chassis	zero-length-string (read-write)
Container	zero-length-string (read-write)
Module	zero-length-string (read-write)
Splitter card	zero-length-string (read-write)
DSL Port	zero-length-string (read-only)
T1/E1 port	zero-length-string (read-only)
OC3 port	zero-length-string (read-only)
Console port	zero-length-string (read-only)
Ethernet Mgmt port	zero-length-string (read-only)
Temp Sensor	zero-length-string (read-only)
Fan	zero-length-string (read-only)
PowerSupply	zero-length-string (read-only)
Management Plane PLD	zero-length-string (read-only)

7.1.2.1.16 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):

The following table shows the entPhysicalIsFRU designated for the entities.

Table 1-75. entPhysicalIsFRU (1 of 2)

Entity	entPhysicalIsFRU
Chassis	TRUE
Container	TRUE
Module	TRUE

Table 1-75. entPhysicalIsFRU (2 of 2)

Entity	entPhysicalIsFRU
Splitter card	TRUE
DSL Port	FALSE
T1/E1 port	FALSE
OC3 port	FALSE
Console port	FALSE
Ethernet Mgmt port	FALSE
Temp Sensor	FALSE
Fan	FALSE
PowerSupply	FALSE
Management Plane PLD	FALSE

7.1.2.2 entityLogical Group.

The entLogicalTable { entityLogical 1} described below shows the attributes supported for each the physical entities. For the GranDSLAM 3.0 product, there will be 1 row associated to the MCP if it is physically coexisting in the same chassis with the SCP card. The SCP will not be able to manage the MCP, thus most of the objects in this table will provide default values.

Table 1-76. entLogicalEntry {entLogicalTable 1}

Object	OID	Syntax	Access	Status	Sup-ported
entLogicalIndex	{entLogicalEntry 1}	INTEGER	not-accessible	current	Yes
entLogicalDescr	{entLogicalEntry 2}	SnmpAdminString	read-only	current	Yes
entLogicalType	{entLogicalEntry 3}	AutonomousType	read-only	current	Yes
entLogicalCommunity	{entLogicalEntry 4}	OCTET STRING	read-only	deprecated	No
entLogicalTAddress	{entLogicalEntry 5}	TAddress	read-only	current	Yes
eentLogicalTDomain	{entLogicalEntry 6}	ITDomain	read-only	current	Yes
entLogicalContextEngineID	{entLogicalEntry 7}	SnmpAdminString	read-only	current	Yes
entLogicalContextName	{entLogicalEntry 8}	SnmpAdminString	read-only	current	

7.1.2.2.1 entLogicalIndex.

The value of this object will identify the entPhysicalIndex associated to the MCP card.

7.1.2.2.2 entLogicalDescr.

The value of this object will correspond to the sysDescr of the MCP card.

7.1.2.2.3 entLogicalType.

The value of this object will be mib-2.

7.1.2.2.4 entLogicalTAddress.

The value of this object will be the IpAddress and UDP port used to access the MCP card.

7.1.2.2.5 entLogicalIPDomain.

The value of this object will be the kind of transport service by which the MCP card can receive management traffic. This value will be snmpUDPDomain.

7.1.2.2.6 entLogicalContextEngineID.

The value of this object will be the null-string.

7.1.2.2.7 entLogicalContextName.

The value of this object will be the null-string.

7.1.2.3 entAliasMappingTable.

The entAliasMappingTable {entityMapping 2} will be supported for all the ports specified in the entPhysicalTable to allow mapping between ifIndexes and the corresponding entPhysicalIndex. The entAliasLogicalIndexOrZero shall be zero. The entAliasMappingIdentifier is a pointer that points to the ifTable row.

Table 1-77. entAliasMappingEntry {entAliasMappingTable 1}

Object	OID	Syntax	Access	Status	Sup-ported
entAliasLogicalIndexOrZero	{entAliasMappingEntry 1}	INTEGER	not-accessible	current	Yes
entAliasMappingIdentifier	{entAliasMappingEntry 2}	RowPointer	read-only	current	Yes

7.1.2.4 entPhysicalContainsTable.

The entPhysicalContainsTable {entityMapping 3} will be supported to expose the container/”containee” relationships between physical entities.

Table 1-78. entPhysicalContainsEntry {entPhysicalContainsTable 1}

Object	OID	Syntax	Access	Status	Sup-ported
entPhysicalChildIndex	{entPhysicalContainsEntry 1}	PhysicalIndex	not-accessible	current	Yes

Valid Object Value(s):

The following table shows the entPhysicalChildIndex values for the different entities.ss=[1..20].

Table 1-79. entPhysicalChildIndex

Entity	entPhysicalIndex	entPhysicalChildIndex	Semantic
Chassis	0001000000	0001ss0000	A chassis contains 'ss' containers, ss=[1..20].
		0001ss010x	A chassis contains a sensor 'x', x=[1..3].
		0001ss011x	A chassis contains a fan "x", x=[1..1]
		0001ss013x	A chassis contains a powersupply "x", x=[1..2]
Container	0001ss0000	0001ss000y	A container 'ss' contains module 'y', ss=[1..20], y=[1..3]
Module	0001ss0001	0001ss1yyy	A module (e.g portcard) contains up to 24 DSL ports, where yyy=[1..24]
		0001ss1801	A module entity contains a console port.
		0001001802	A module entity contains a ethernet management port.
		0001ss0121	A module entity contains a Mgmt Plane PLD

7.1.2.5 entityGeneral Group.

This group will be supported as specified in RFC 2737.

7.1.2.6 entityNotifications Group

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. This trap will be generated as specified in the RFC 2737.

7.2 Paradyne Entity MIB extension (mpe_dslam.mib)

The following MIB table will extend the entityMIB (RFC 2737).

Table 1-80. Paradyne Entity MIB Extension

Table	Comments
mpeEntPhysicalExtEntry	Supported as is

The following subsections highlights special cases and/or limitations on the support of some objects in this table.

7.2.1 mpeEntPhysicalExtAdminStatus

This object allows managers to set the administrative status of each entity. All the entities dynamically discovered by the device will have an adminstatus set to 'up' by default. This object will be supported read-only.

7.2.2 mpeEntPhysicalExtOperStatus

This is a read-only object that allows managers to determine the operational status of the entities in the managed device. The possibles values are 'up', 'down' and 'testing'.

7.3 Entity Sensor MIB (RFC 3433):

This MIB has only one group (entitySensorValueGroup) which consists of a single (read-only) table that will be supported as indicated in the following table. There will be as many entries as sensors. For the GranDSLAM 3.0 product, there will be three temperature sensors

Table 1-81. Entity Sensor MIB

Table	Comments
entPhySensorEntry	Only the objects listed below will be supported

7.3.1 entPhySensorType.

The value for this object corresponds to the measurement data type. The value of this object will be celsius (8).

7.3.2 entPhySensorValue.

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

7.3.3 entPhySensorOperStatus.

The operational status of the sensor. The following status will be reported: ok (1), unavailable (2), nonoperational (3).

7.3.4 entPhySensorValueTimeStamp.

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

7.3.5 entPhySensorValueUpdateRate.

The value of this object is an indication of the frequency that the agent updates the associated entPhySensorValue object. The value to be returned will be zero. This value has a special meaning in the mib.

7.4 Definitions for the IEEE 802.3 Medium Attachment Units (MAUs) (RFC 2668):

RFC 2668 is used for accessing information about of ethernet interface. The following objects are supported.

7.4.1 ifMauTable:

Table of descriptive and status information about MAU(s) attached to an interface. The following objects in this table are supported.

Table 1-82. ifMauTable

Table	Object	Type	Access	Supported
ifMauTable	ifMauIfIndex	Integer32	RO	Yes
	ifMauIndex	Integer32	RO	Yes
	ifMauType	OBJECT IDENTIFIER	RO	Yes
	ifMauStatus	INTEGER	RW	Yes
	ifMauMediaAvailable	INTEGER	RO	No
	ifMauMediaAvailableStateExits	Counter32	RO	No
	ifMauJabberState	INTEGER	RO	No
	ifMauJabberingStateEnters	Counter32	RO	No
	ifMauFalseCarriers	Counter32	RO	No
	ifMauTypeList	Integer32	RO (deprecated)	No
	ifMauDefaultType	OBJECT IDENTIFIER	RW	Yes
	ifMauAutoNegSupported	TruthValue	RO	Yes
	ifMauTypeListBits	BITS	RO	Yes

7.4.1.1 ifMauIfIndex (RO):

This variable uniquely identifies the interface to which the MAU described by this entry is connected. This value corresponds to the ifIndex of the interface.

7.4.1.2 ifMauIndex (RO):

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

7.4.1.3 ifMauType (RO):

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.

The set of valid MAU object identifier types are as defined by the dot3MauType object as:

dot3MauTypeAUI, dot3MauType10Base5, dot3MauTypeFoirl, dot3MauType10Base2, dot3MauType10BaseT, dot3MauType10BaseFP, dot3MauType10BaseFB, dot3MauType10BaseFL, dot3MauType10Broad36, dot3MauType10BaseTHD, dot3MauType10BaseTFD, dot3MauType10BaseFLHD, dot3MauType10BaseFLFD, dot3MauType100BaseT4, dot3MauType100BaseTXHD, dot3MauType100BaseTXFD,

dot3MauType100BaseFXHD, dot3MauType100BaseFXFD, dot3MauType100BaseT2HD, dot3MauType100BaseT2FD, etc.

7.4.1.4 ifMauStatus (RO):

The current state of the MAU.

7.4.1.5 ifMauDefaultType (RW):

This object identifies the default administrative baseband MAU type, to be used in conjunction with the operational MAU type denoted by ifMauType.

7.4.1.6 ifMauAutoNegSupported (RO):

This object indicates whether or not auto-negotiation is supported on this MAU.

7.4.1.7 ifMauTypeListBits (RO):

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 above.

All other objects in this table are not supported.

7.5 IP Forwarding Table (RFC 2096)

Table 1-83. IP Forwarding Table

Table	Comments
ipCidrRouteNumber	Indicates the actual number of routes in the table
ipCidrRouteEntry	Supported as is

7.6 Paradyne PDN-MGMT-IP-MIB.

This enterprise MIB contains some objects for configuration and management of Classical IP ports (also called IP interfaces) that are used for managing a DSLAM device. Using the objects in this mib, we can managed the DSLAM using either the IP-over-Ethernet or the IP-over-ATM models. The following MIB objects are supported to accomplish that.

Table 1-84. PDN-MGMT-IP-MIB

Table	Object	Type	Supported
pdnMgmtIpPortTable	pdnMgmtIpPortIndex	InterfaceIndex	Yes
	pdnMgmtIpAddress	IpAddress	Yes
	pdnMgmtIpNetMask	IpAddress	Yes
	pdnMgmtIpEthGateway	IpAddress	Yes
	pdnMgmtIpPhysAddress	PhysAddress	Yes
	pdnMgmtIpConfigMode	INTEGER	Yes
	pdnMgmtBootIfIndex	InterfaceIndex	Yes
	pdnMgmtBootVpi	ATMVpIdentifier	Yes
	pdnMgmtBootVci	ATMVcIdentifier	Yes
	pdnMgmtIpAdminStatus	INTEGER	Yes
pdnMgmtAtmInvArpEntry	pdnMgmtAtmIfIndex	InterfaceIndex	Yes
	pdnMgmtAtmVpi	ATMVpIdentifier	Yes
	pdnMgmtAtmVci	ATMVcIdentifier	Yes
	pdnMgmtIpPortIfIndex	InterfaceIndex	Yes
	pdnMgmtNextHopIp	IpAddress	Yes
	pdnMgmtAtmInvArpRowStatus	RowStatus	Yes
	pdnMgmtIpDefaultRouter	IpAddress	Yes

7.6.1 pdnMgmtIpPortTable

This is the IP Port Table. There is one entry per IP port/interface in the system.This table contains the configuration details of each Ip port. This table is indexed on the interface index of the IP port.

7.6.1.1 pdnMgmtIpPortIndex (NA)

The index assigned to the IP Port. Currently, no Ip port is represented in the ifTable of RFC 2863. The following table lists the interface indexes to appear on the IpPortTable.

Table 1-85. pdnMgmtIpPortIndex list

Interface	ifInterface Index (ifIndex)	Interface Descr
IP Port over Ethernet	001002802	“Ip Port over Ethernet”
IP Port over ATM-1	001004803	“Ip Port over ATM-1”
IP Port over ATM-2	001004804	“Ip Port over ATM-2”

7.6.1.2 pdnMgmtIpAddress (RW).

The IP address assigned to this port. If the pdnMgmtIpConfigMode is set to modes other than 'manual', then the MAX-ACCESS for this object is read-only and the value of the object represents the actual IP address assigned to the port by the DHCP client (or 0.0.0.0, if none assigned). Changes to this object could disrupt data flow through the port.

7.6.1.3 pdnMgmtIpNetMask (RW).

The IP subnet mask assigned to this port.

The MAX-ACCESS for this object behaves the same as the pdnMgmtIpAddress explained above. Changes to this object could disrupt data flow through the port.

7.6.1.4 pdnMgmtIpEthGateway (RW).

The IP Gateway for an ethernet based IP port.

This object is supported only on IP ports that are configured to run over ethernet type of media. This object would be set to 0.0.0.0 for other types of interfaces and is not valid.

7.6.1.5 pdnMgmtIpPhysAddress (RW).

The physical address associated with the media.

For IP Ports with Ethernet type of media, the MAX-ACCESS for this object is restricted to read-only. In such cases, this object would reflect the MAC address of the underlying ethernet data link. For IP Ports that run over ATM PVCs, this object is writable.

7.6.1.6 pdnMgmtIpConfigMode (RW).

The IP configuration mode for the port. The possible values are: manual(1), dhcp(2).

In "manual" mode, it is up to the user to provide the IP address and IP subnet mask of the port.

In "dhcp" mode, the respective protocols are resposible to obtain and populate the IP address and IP subnet mask for the port. The default value for all the interfaces is dhcp(2).

Whenever the config mode is changed from "manual" mode, the IP address and the IP subnet mask for the port would automatically be reset to 0.0.0.0. External Manager stations SHOULD NOT attempt to write the IP address or the IP subnet mask of the port when this object is not in "manual" mode.

7.6.1.7 pdnMgmtBootIfIndex (RW)

The interface index of the ATM interface to be used by the DHCP client, if the pdnMgmtIpConfigMode is set to "dhcp" respectively. For the Ethernet's IP port, this object is 0.

7.6.1.8 pdnMgmtBootVpi (RW)

The VPI of the ATM PVC to be used by the DHCP client, if the pdnMgmtIpConfigMode is set to "dhcp" respectively.

7.6.1.9 pdnMgmtBootVci (RW)

The VCI of the ATM PVC to be used by the DHCP client, if the pdnMgmtIpConfigMode is set to "dhcp" respectively.

7.6.1.10 pdnMgmtIpAdminStatus (RW)

The status of the IP Port (interface). Possible values are up(1), down(2).

7.6.2 pdnMgmtAtmInvArpTable.

This table contains one row per Inverse ATM ARP entry in the system. This table maps the <ifIndex, vpi, vci> index to corresponding <ipPortIndex, remoteIp> of the pdnMgmtIpPortTable in 3.30.1.

7.6.2.1 pdnMgmtAtmIfIndex (NA)

The interface index associated with the ATM PVC.

7.6.2.2 pdnMgmtAtmVpi (NA)

The VPI associated with the ATM PVC.

7.6.2.3 pdnMgmtAtmVci (NA)

The VCI associated with the ATM PVC.

7.6.2.4 pdnMgmtIpPortIfIndex (RC)

The Interface Index of the IP port to which this inverse ARP entry is associated with. Refer to the pdnMgmtIpPortTable for configuration of IP Ports.

7.6.2.5 pdnMgmtNextHopIp (RC)

The IP address of the unit at the other end of this ATM PVC. This represents the IP Gateway equivalent for IP traffic on this specific ATM PVC.

7.6.2.6 pdnMgmtAtmInvArpRowStatus (RC)

This object is used to create/delete the ATM inverse ARP entry in this table.

pdnMgmtIpDefaultRouter (RW)

This object identifies the default IP router to use for IP packets with no other known route.

An IP address of 0.0.0.0 would disable and delete the default route from the system.

7.7 SNMPv3.

7.7.1 Framework and Applications.

Based on current requirements for the GranDSLAM 3.0 project, this is the level of support desired:

Command Generators: It is not required that our DSLAM products generate SNMP messages. However, it may not be possible to turn this feature off from the Emanate/Lite agent configuration.

Command Responders: This is the primary function of our DSLAM management entity. Thus, this will be implemented as indicated in RFC 2573.

Notification Originators: Initiating traps is a primary function of the SCP management entities. The INFORM operation (Confirmed notifications) is currently not required in our products and will not be implemented.

Notification Receivers: It is not required that our DSLAM products be able to receive and process notifications from other Entities, thus this application will not be implemented.

Proxy Forwarders: It is not required that our GranDSLAM 3.0 product act as a proxy since we will have only one SNMP engine running. The implementation of this application is optional as per the standard.

7.7.2 Supporting MIBs.

7.7.3 SNMP-MPD-MIB (RFC 3412)

This MIB provides statistics for message processing. The following scalar MIB objects will be supported to provide statistics about invalid or unknown conditions in the message processing:

7.7.3.1 snmpUnknownSecurityModels

7.7.3.2 snmpInvalidMsgs

7.7.3.3 snmpUnknownPDUHandlers

7.7.4 SNMP Framework MIB (RFC 3411)

In addition to defining new textual conventions, this MIB defines the following scalar objects that will be supported in the SCP:

7.7.4.1 snmpEngineID:

There will be a unique identifier for the SNMP engine running in the device. It will be set up by following one of the schemes proposed in the RFC 3411. That is, it will use the private enterprise number assigned to Paradyne (1795) by the Internet Assigned Numbers Authority and the MAC address of the card (SCP) running the agent to generate a unique identifier for the Engine.

For example the MAC address of my SCP card is ’00-E0-39-C1-CD-16’ will result in the following snmpEngineID:

80 00 07 03 03 00 E0 39 C1 CD 16

See RFC 3411 for details on how this is formatted.

7.7.4.2 snmpEngineBoots:

Number of times this entity has been reboot. This number must be preserved upon reset of the entity.

7.7.4.3 snmpEngineTime:

Actual sysUpTime since last reboot.

7.7.4.4 snmpMaxMessageSize: Based on configuration of the Emanate/Lite agent.

7.7.4.5 SNMP-TARGET-MIB (RFC 2573)

The SCP will use the SNMP-TARGET-MIB exclusively to configure trap managers. The following MIB objects contained in SNMP-TARGET-MIB will be **supported** to configure trap managers.

7.7.4.5.1 snmpTargetSpinLock:

It is used to control multiple simultaneous access to the target table. Users must follow the procedure in RFC 2573 to add/modify entries in this table.

7.7.4.5.2 snmpTargetAddrTable:

It is used to configure transport addresses to be used in the generation of SNMP notification messages.

7.7.4.5.3 snmpTargetParmsTable:

It is used to configure target parameters (message processing level, security model, etc.) that will be used in the generation of SNMP notification messages. This table will be supported read-only with default entries as indicated in section 7.7.4.6.4 and 7.7.4.6.5

7.7.4.5.4 snmpUnavailableContexts and snmpUnknownContexts:

Statistics.

These tables will be stored in nonvolatile memory.

7.7.4.6 SNMP-NOTIFICATION-MIB (RFC 3413)

This MIB allows the GranDSLAM 3.0 agent to configure notification recipients. The MIB is composed of the following tables:

7.7.4.6.1 snmpNotifyTable:

It contains entries that allow the GranDSLAM 3.0 to select the target entry in the snmpTargetAddrTable that should be used when generating notification. Confirmed (NOTIFY) and Unconfirmed notifications will be supported without encryption of the data.

7.7.4.6.2 snmpNotifyFilterTable:

It contains filter profiles. Filter profiles are used to determine whether particular management targets should receive certain notifications. This filtering feature will not be supported.

7.7.4.6.3 snmpNotifyFilterProfileTable:

Filtering of traps will not be supported.

7.7.4.6.4 SNMPv1 traps - Default values:

By default, the GranDSLAM 3.0 agent will generate SNMPv1 traps. The notification message will not be privatelly protected (no encryption). There will be a corresponding entry in the snmpTargetParamsTable to be used when generating SNMPv1 traps. This entry will have the following default values that can not be removed from the table:

snmpTargetParamsName='defaultV1traps',

snmpTargetParamsMPModel='SNMPv1 (0)',

snmpTargetParamsSecurityModel='SNMPV1 (1)',

snmpTargetParamsSecurityLevel='noAuthNoPriv (1).

7.7.4.6.5 SNMPv2 traps - Default values:

Similarly, 'defaultV2traps' will be defined to be used when generating SNMPv2 traps.

The following default values will be associated to SNMPv2 traps:

snmpTargetParamsName='defaultV2traps',

snmpTargetParamsMPModel='SNMPv2 (1)',

snmpTargetParamsSecurityModel='SNMPV1 (1)',

snmpTargetParamsSecurityLevel='noAuthNoPriv (1).

The trap version can be selected by specifying the desire choice (defaultV1traps, defaultV2traps) in the snmpTargetAddrParams object of the trap manager configuration (snmpTargetAddrTable).

7.7.5 User-based Security Model (USM) RFC 3414

RFC 3414 discusses the “User-based security model” for SNMPv3. It defines the elements of procedure for providing SNMP message-level security. The mechanisms to be implemented related to this feature are Discovery and Timeliness, Authentication, Privacy and Key management.

The product will support the HMAC-MD5-96 and the HMAC-SHA-96 protocols for authentication and the CBC-DES Symmetric Encryption Protocol for Privacy.

7.7.5.1 Supporting MIBs.

The following statistics MIB objects will be supported:

7.7.5.1.1 Statistics.

usmStatsUnsuppportedSecLevels, usmStatsNonInTimeWindows, usmStatsUnknownUserNames, usmStatsUnknownEngineIDs, usmStatsWrongDigests, usmStatsDecryptionErrors.

7.7.5.1.2 SNMPv3 users.

7.7.5.1.2.1 usmUserTable.

Will be supported to maintain authentication and privacy information for each user. The engineID and the userName index the table. For the GranDSLAM 3.0 product, all entries will have the same local engineID.

Because new SNMPv3 users can be added to this table only by cloning it from an existing entry, we need an initial entry to start with. The initial entry will be based on the password of our default userID. This will be done only the first time SNMPv3 is turned on (’snmpV3-encryption’ option is selected).

This initial user/password is run through an algorithm based on the HMAC-MD5-96 (default algorithm for authentication) and converted in what is called a **localized key**. This procedure is stardarized in RFC 3414.

Remote entities (for example, EMS) must obtained the **same** value of the **localized key** to start with.

Once the initial entry is created, clients (EMS, TL-1, Web, etc.) will used a standarized procedure in RFC 3214 to clone new users from the existing entries in the usmUserTable. No other mibs are involved in creating SNMPv3 users.

According to requirements, the SNMPv3 users to be configured will always have AuthPriv as the securityLevel, that is both authentication and privacy (encryption) turn on. SecurityLevel of NoAuthNoPriv or AuthNoPriv **will not be supported** for these users.

7.7.5.1.2.2 usmUserSpinLock.

This object will be supported to coordinate set operations to the usmUserTable.

7.7.6 View-based Access Control (VACM)

RFC 3415 discusses the “View-based Access Control Model” for SNMPv3. The GranDSLAM 3.0 agent will create default entries in the neccesary tables to be commonly used between v1/v2c/v3 SNMP users.

7.7.6.1 Supporting MIBs

RFC 3415 defines several tables to be used to determine if a SNMP operation (get, getnext, getbulk, set or notification) is allowed to access certain managed objects.

7.7.6.1.1 vacmContextTable.

It defines the locally available contexts. This table is read-only and can not be configured by SNMP.

7.7.6.1.2 vacmSecurityToGroupTable.

It maps a securityModel and securityName to a groupName. Supported read-only with default values.

7.7.6.1.3 vacmAccessTable.

It maps a groupName, context and securityInformation into a MIB view. Supported read-only with default values.

7.7.6.1.4 vacmViewTreeFamilyTable.

It defines whether an OBJECT IDENTIFIER can be accessed for a given MIB view or not. Supported read-only with default values.

7.7.7 SNMPv3 Coexistence.

RFC 2576 discusses the “Coexistence between Version 1, Version 2 and Version 3 of the Internet-standard Network Management Framework”. This table will be used to configure community strings for v1/v2c clients. This table has no significance and thus, it is not used for SNMPv3 clients.

By default, the SCP agent will configure a default community string ’public’ with read/write access level; which uses the v1 modeling processing and dispatching [snmpV1 (1)] as well as the community-based security model.

7.7.7.1 Supporting MIBs

7.7.7.1.1 SNMP-COMMUNITY-MIB.

7.7.7.1.1.1 snmpCommunityTable.

It provides a mapping of a community string to a securityName (for example, snmpV1 (1) or snmpV2 (2)), contextEngineID, and contextName.

If configured, this table also allows the source address to be validated which provides some level of security for SNMPv1 and SNMPv2c environments. This source address validation will not be supported.

Because the GranDSLAM 3.0 will be configured as an SNMPv1, and entry in this table will exist by default with a communityName of “public” and the source address validation will be turned off (snmpCommunityTransportTag = empty-string).

A TL1 command or Web interface option will be required to enable/disable different versions of SNMP or ’none’ (the options are: ’None’, ’v1/v2c’ ’v3 with encryption’. An enterprise MIB might be required to handle this option. These commands will internally translate into additionsdeletions to the respective tables (snmpCommunityTable for v1/v2c users, usmUserTable for v3 users).

1) When ’v1/v2c’ is selected, users in usmUserTable will be set to inactive.

2) When ’v3-encryption’ is selected, users in entCommunityTable will be set to inactive.

3) When ’none’ is selected, users in both usmUserTable and entCommunityTable will be set to inactive.

7.8 Link Fault Management.

This sections describes the mibs associated with the Automatic Protection Switching (APS), Manual Dual Link Load Sharing (DLLS) and Redundancy Features.

7.8.1 Automatic Protection Switching (APS) - draft-ietf-atommib-sonetaps-mib-xx.txt

This MIB objects are used to control and manage SONET linear APS architectures. The generation of APS notifications is enabled by default (apsNotificationEnable).

The object pdnLinkFaultMgmtApsSelection is used to enable/disable APS in the device. By default this object is disabled.

Table 1-86. pdnLinkFaultMgmtApsSelection

Table	Comments
apsConfigEntry	Supported as is except for the special cases indicated here.
apsStatusEntry	Supported as is
apsMapEntry	Supported as is
apsChanConfigEntry	Supported as is except for the special cases indicated here.
apsCommandEntry	Supported as is
apsChanStatusEntry	Supported as is
apsNotificationEnable	Supported as is

The following subsections highlights special cases and/or limitations on the support of some objects in these tables.

7.8.1.1 apsConfigMode.

This object allows configuration of the APS mode. Only the following two values are to be supported: onePlusOne(1) and oneToN (2).

7.8.1.2 apsConfigRowStatus

This object is supported read-write rather than read-create. Thus, no new rows can be added to this table.

7.8.1.3 apsChanConfigRowStatus

This object is supported **read-write** rather than read-create. Thus, no new rows can be added to this table.

7.8.1.4 apsConfigExtraTraffic.

This object enables or disables the transfer of extra traffic on the protection channel in a 1:n architecture. This object will be supported read-only**.**

7.8.2 Link Load Sharing (LLS) - (PDN-LINK-LOAD-SHARING-MIB.mib)

The object pdnLinkLoadSharingSelection is used to enable Load Sharing in the device. By default Load Sharing is disabled**.**

Table 1-87. Link Load Sharing

Table	Comments
pdnLLSConfigEntry	Supported as is
pdnLLSMapEntry	Supported as is

7.8.2.1 pdnLLSConfigGroupName.

This object contains a text describing a group configured for load sharing.

7.8.2.2 pdnLLSConfigSelection.

This object allows managers to enable and disable the link load sharing on this group.The default value for this object is **disabled**.

7.8.2.3 pdnLLSMapGroupName.

This object is used in the pdnLLSMapEntry to map uplink interfaces and load sharing groups. This Group Name corresponds to one of the groups defined in the pdnLLSConfigTable.

7.8.3 Enterprise link failure management- (PDN-LINK-FAULT-MGMT-MIB.mib)

The object pdnLinkFaultMgmtSwitchoverSelection is used to enable and disable the Switchover due to link failure feature in this device. By default, switchover is **disabled**. This feature can not be enabled if APS is enabled.

Table 1-88. pdnLinkFaultMgmtSwitchoverSelection

Table	Comments
pdnLinkFailureConfigEntry	Supported as is

7.8.4 Entity redundancy - (PDN-ENTITY-REDUNDANCY-MIB.mib)

The object **pdnEntityRedundancySelection** is used to enable and disable the redundancy feature. By default, this object is **disabled**.

Table 1-89. pdnEntityRedundancySelection

Table	Comments
pdnRedunCmdEntry	Supported as is
pdnRedunStatusEntry	Supported as is

7.8.4.1 pdnRedunCommand.

This MIB object allows managers to command an action to an entity module (given by the entPhysicalIndex), configured for redundancy. The supported commands are switch(2) and forceswitch (3). The value noCmd (1) will be returned when no previous command had been issued. Users should specify the slot-based entPhysicalIndex (for example, 1190001) corresponding to the entity that will take the action.

7.8.4.2 pdnRedunAlarmStatus.

This MIB object displays the status of any outstanding major alarms in an entity of a redundant configuration. If no major alarms are present, this object will return 0. This MIB object is a Bitmap to report multiple major alarm conditions. This MIB object is indexed by the slot-based entPhysicalIndex (for example, 1190001).

7.8.4.3 pdnRedunEntityState.

This MIB object displays the state of each entity module configured for redundancy. The possible states reported are: activeState (1), activeAlarmState (2), standbyState (3), standbyAlarmState (4). This MIB object is indexed by the slot-based entPhysicalIndex (for example, 1190001).

7.8.4.4 pdnRedunNotificationEnable.

This MIB object provides the ability to enable and disable the generation of traps related to redundancy in the specified entity. The default value of this object is disable. This MIB object is indexed by the slot-based entPhysicalIndex (for example, 1190001).

8. Timing and Clocking

8.1 Paradyne mpe_Config MIB (mpe_Config.mib)

The tabular object mpeDevConfigClockSrcEntry defined in mpe_Config.mib provides the capability of choosing network timing reference source on the GranDSLAM.

Table 1-90. mpeDevConfigClockSrcEntry

Table	Object	Type	Access	Supported
mpeDevConfigClockSrcEntry	mpeDevCfgClkWhichSrc	INTEGER	read-write	Yes
	mpeDevCfgClkSource	INTEGER	read-write	Yes
	mpeDevCfgClkIfIndex	Integer32	read-write	Yes
	entPhysicalIndex	PhysicalIndex	not-accessible	Yes

The mpeDevCfgClkWhichSrc object takes the following two values:

- primary(1), this chooses the primary master clock source,
- secondary(2), this chooses the secondary master clock source.

Write access to this object is not supported. Only “primary(1)” is granted.

The mpeDevCfgClkSource 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. It contains the following values:

- internal(1), Master clock is the internal clock,
- external(2), Master clock is the external clock source,

Write access to this object is supported; Only “internal(1)” and “interface (3)” are granted.

The mpeDevCfgClkIfIndex is used to select the interface to be used as the source for the master clock for the device, if mpeDevCfgClkSource is set to interface(3).

The value of entPhysicalIndex is the physical index of this card itself, which uniquely identifies the stackable unit.

8.2 Paradyne Config MIB (pdn_Config.mib)

Table 1-91. pdn_Config.mib

Table	Object	Type	Access	Supported
devConfigTestTimer				Yes
	devConfigTestTimeout	INTEGER	read-write	Yes
	devConfigTestDuration	INTEGER	read-write	Yes

The pdn_config MIB is specifically used to configure overall test timeouts. The objects supported are:

8.2.1 devConfigTestTimeout:

The valid values are :

disable(1) - Tests will not be terminated based on a timer and

enable(2) - Tests will be 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.

8.2.2 devConfigTestDuration

This object specifies the duration (in seconds) that a test will be allowed to run before it is automatically terminated. The range of possible values are 0-7200 seconds with increament of 60 Sec. The default value is 600 Seconds (10 Minutes). Tests will only be terminated if devConfigTestTimeout is set to enable.

8.3 Paradyne Time Mib (pdn_time.mib)

The following objects are supported for GranDSLAM 3.0.

Table 1-92. pdn_time.mib

Object	Type	Access	Supported
devDateAndTime	DateAndTime	read-write	Yes
devNTPServerIP	IpAddress	read-write	Yes
devNTPMode	NTPMode	read-write	Yes
devNTPSynchronised	INTEGER	read-write	Yes
devNTPEnable	INTEGER	read-write	Yes
devNTPOffsetFromUTC	INTEGER	read-write	Yes

8.3.1 devDateAndTime (RW):

This objects displays the date and time on the entity. It allows an NMS to set the date and time in the appropriate timezone.

8.3.2 devNTPServerIP (RW):

This Objects allows an NMS to configure theNTP server IP address.

8.3.3 devNTPMode (RW):

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 multicast(3).

8.3.4 devNTPSynchronised (RW):

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.

8.3.5 devNTPEnable (RW):

.Indicates whether or not the local clock is synchronized via NTP/SNTP. 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).

8.3.6 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 (for example, offsets for local times west of the zero meridian to the international date line are negative). Default value is 0 (that is, at UTC).

9. Access Control and Security.

9.1 Paradyne DSLAM System Mib [pdn_dslam.mib] (pdn-common 24) .

Support for the pdn_dslam MIB is provided to enable an NMS access certain objects that are specific to this product.

Table 1-93. pdn_dslam.mib (Objects Supported)

Table	Object	Type	Supported
sysDevUserAccountTable	sysDevUserAccountUserId	DisplayString	Yes
	sysDevUserAccountPrivilege	INTEGER	Yes
	sysDevUserAccountUserPassword	DisplayString	Yes
	sysDevUserAccountAccessPartition	DisplayString	Yes
	sysDevUserAccountRowStatus	RowStatus	Yes

The various objects are described in the above table.

9.1.1 sysDevUserAccountTable:

This table contains the user accounts.The various supported objects are :

9.1.1.1 sysDevUserAccountUserId (NA):

This object corresponds to the login ID of the user account. The login shall be from 1 to 10 printable characters.

9.1.1.2 sysDevUserAccountPrivilege (RC):

Valid values are operator (1), administrator (2), maintenance (3), provisioning (4), manufacturing (5). This object corresponds to the access privilege of the user account.

9.1.1.3 sysDevUserAccountUserPassword (RC):

This object corresponds to the password of the user account. The password shall be from 8 to 10 printable characters.It will not be possible delete that last Password with Administrator access.[The user must create a new one prior to deleting the only administrator].

9.1.1.4 sysDevUserAccountAccessPartition (RC):

This object corresponds to the access partition of the user account. The default value of this object is 'all'.

9.1.1.5 sysDevUserAccountRowStatus (RC):

This object corresponds to create or delete a row in sysDevUserAccountTable. A maximum of six UserID/Password combinations can be created. By default one entry is created with UserID=SUPERUSER and Password=ASN#1500.

9.2 Paradyne Security MIB (pdn_Security.mib):

The pdn_security MIB is used for enabling applications control security aspects of the unit for the purpose of Telnet, FTP, and so on. The various objects supported are as follows:

Table 1-94. pdn_Security.mib

Table	Object	Type	Supported
scalar object	devSecurityMgrValidation	INTEGER	Yes
scalar object	devSecurityMgrMaxNumber	INTEGER	Yes
scalar object	devSecurityMgrCurrentNumber	INTEGER	Yes
securityMgrTable	securityMgrIpAddress	IpAddress	Yes
	securityMgrSubnetMask	IpAddress	Yes
	securityMgrSnmpAccess	INTEGER	Yes
	securityMgrTelnetAccess	INTEGER	Yes
	securityMgrFtpAccess	INTEGER	Yes
	securityMgrTrapAccess	INTEGER	Yes
	securityMgrRowStatus	RowStatus	Yes

9.2.1 devSecurityMgrValidation (RW):

This object is used to either enable or disable SNMP source IP validation. When enabled the first check on any incoming PDU is based on the source IP address of the datagram. If the IP address is configured in the securityMgrTable 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. By default, this object is set to disable(1).

9.2.2 devSecurityMgrMaxNumber (RO):

The maximum number of managers allowed in the devSecurityMgrTable.

9.2.3 devSecurityMgrCurrentNumber (RO):

The current number of managers in the devSecurityMgrTable..

9.2.4 securityMgrTable:

A list of the security manager entries. The various objects supported are:

9.2.4.1 securityMgrIpAddress (NA):

This is used to specify the IP address that identifies the SNMP manager(s) that are authorized to send SNMP messages.

9.2.4.2 securityMgrSubnetMask (NA):

This object is used to specify the subnet mask.

9.2.4.3 securityMgrSnmpAccess (RW):

This is used to specify the snmp access to the device. This object is tied with the securityMgrTelnetAccess and securityMgrFtpAccess. That is, changing the SnmpAccess will also change those two objects.

9.2.4.4 securityMgrTelnetAccess(RW):

This is used to specify the telnet access to the device. This object is tied with the securityMgrSnmpAccess and securityMgrFtpAccess. That is, changing the TelnetAccess will also change those two objects.

9.2.4.5 securityMgrFtpAccess (RW):

This is used to specify the ftp access to the device. This object is tied with the securityMgrTelnetAccess and securityMgrSnmpAccess. That is, changing the FtpAccess will also change those two objects.

9.2.4.6 securityMgrTrapAccess (RW):

This is used to specify the trap access to the device. By enabling trap access, the device will automatically configure this Manager as a recipient of traps. This object is deprecated now.

9.2.4.7 securityMgrRowStatus (RW):

This is used to create and remove entries to this table.

9.3 Paradyne Control MIB (mpe_Control.mib)

Table 1-95. mpe_Control.mib (1 of 2)

Table	Object	Type	Supported
mpeDevControlTable			Yes
	mpeDevControlReset(1)	ResetStates	Yes
	entPhysicalIndex	PhysicalIndex	Yes
mpeDevFileXferConfigTable			Yes
	entPhysicalIndex	Integer32	Yes
	mpeDevFileXferFileName(1)	DisplayString	Yes
	mpeDevFileXferCopyProtocol(2)	INTEGER	Yes
	mpeDevFileXferFileType(3)	INTEGER	Yes
	mpeDevFileXferServerIpAddress(4)	IpAddress	Yes
	mpeDevFileXferUserName(5)	DisplayString	Yes
	mpeDevFileXferUserPassword(6)	DisplayString	Yes
	mpeDevFileXferOperation(7)	INTEGER	Yes
	mpeDevFileXferPktsSent(8)	Counter32	Yes
	mpeDevFileXferPktsRecv(9)	Counter32	Yes
	mpeDevFileXferOctetsSent(10)	Counter32	Yes
	mpeDevFileXferOctetsRecv(11)	Counter32	Yes
	mpeDevFileXferOwnerString(12)	OCTET STRING	Yes
	mpeDevFileXferStatus(13)	INTEGER	Yes
	mpeDevFileXferErrorStatus(14)	Integer32	Yes
	mpeDevFileXferSendEvent(15)	INTEGER	No
	mpeDevFileXferRowStatus(16)	RowStatus	Yes
	mpeDevFileXferXferTime(17)	TimeTicks	Yes
	mpeDevFileXferFileFormat(18)	INTEGER	Yes
mpeDevControlSelfTestTable			Yes
	entPhysicalIndex	Integer32	Yes
	mpeDevControlExtendedSelfTest	INTEGER	Yes
mpeDevControlTestTable			Yes
	entPhysicalIndex	Integer32	Yes
	mpeDevControlTestType	INTEGER	Yes
	mpeDevControlTestStatus	INTEGER	Yes
	mpeDevControlTestCmd	INTEGER	Yes

Table 1-95. mpe_Control.mib (2 of 2)

Table	Object	Type	Supported
mpeDevFirmwareControlTable			Yes
	entPhysicalIndex		Yes
	mpeDevFirmwareControlIndex(1)	INTEGER	Yes
	mpeDevFirmwareControlRelease(2)	DisplayString	Yes
	mpeDevFirmwareControlOperStatus(4)	INTEGER	Yes
	mpeDevFirmwareControlAdminStatus(5)	INTEGER	Yes

9.3.1 mpeDevControlTable:

This table is indexed on the entPhysicalIndex and contains mpeDevControlReset object which is used to reset the associated entity (for example, card). Writing the value reset (2) to this object initiates a Hardware power-on reset of the entity. Writing the value resetToFactoryDefaults (3) causes the entity to re-configure itself with factory defaults. Writing the value of resetToNewActiveConfig (4) causes a new configuration to be applied without resetting the unit. The value read from this object is noOp(1).

9.3.1.1 mpeDevFileXferConfigTable:

The Paradyne FileXfer Client Config Table is indexed by the entPhysicalIndex. Use of the File Tranfer MIB could be used with other MIBS in the following manner:

a) Use another MIB/means to verify available space /make room for a file to be transfered to this device

b) Use this MIB to download the file.

c) Use another MIB/means to select the file you want to make active if your selecting firmware for example.

d) 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 two operations possible after this are read and 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.

9.3.2 mpeDevFirmwareControlTable :

This table contains a list of the current Firmware Releases and their associated status. Each Firmware Release will be indexed by a number from 1 to N. The user will be 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.The active Firmware will always be the first entry in this table.

10. Status and Alarms.

10.1 Paradyne Syslog MIB (pdn_syslog.mib):

Table 1-96. Syslog MIB

Table	Comments
pdnEntitySyslogEntry	Supported as is
pdnSyslogStatus	Supported as is
pdnSyslogIPAddr	Supported as is
pdnSyslogLevel	Supported as is
pdnSyslogPort	Supported as is
pdnSyslogSeverityThreshold	Supported as is
pdnSyslogRemoteDaemon	Supported as is
pdnSyslogNumOfMsgInTable	Supported as is
pdnSyslogMaxTableSize	Supported as is
pdnSyslogClearTable	Supported as is
pdnSyslogMsgToConsole	Supported as is
pdnSyslogRateLimiting	Supported as is

The pdn_syslog MIB defines a collection of objects for managing syslog messages. The objects are used to configure both syslog functions and syslog daemons. This MIB also allows one to retrieve syslog message from a device.

10.1.1 pdnSyslogStatus (RW):

Valid values are disable (1), enable (2). This object allows user to disable or enable syslog event generation.

10.1.2 pdnSyslogIPAddr (RW):

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.

10.1.3 pdnSyslogLevel:

This object has been deprecated.

10.1.4 pdnSyslogPort (RW):

This value corresponds to a UDP port number to which system events are sent. The default is 514.

10.1.5 pdnSyslogSeverityThreshold (RW):

The valid values are emerg (0), alert (1), critical (2), error (3), warning (4), notice (5), info (6), debug (7). 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 obtain from the RFC 3164, The BSD Syslog Protocol:

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

10.1.6 pdnSyslogRemoteDaemon (RW):

This value allows user to disable or enable syslog messages going to a remote daemon.

10.1.7 pdnEntitySyslogTable (NA):

This table supports retrieval of the syslog messages. It is indexed by the entPhysicalIndex and the pdnSyslogNumber. It allows the clasification of messages per entity (for example, per port card). The table objects are :

10.1.7.1 pdnEntitySyslogNumber (NA):

This is a sequentially increasing index of syslog messages starting at 1. When the table reaches its maximum size (as defined by pdnSyslogMaxTableSize) then the oldest message will be 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).

10.1.7.2 pdnEntitySyslogMessage (RO):

The text of the syslog message.

10.1.8 pdnSyslogNumOfMsgInTable (RO):

This value let the users know how many syslog messages are currently in the syslog table.

10.1.9 pdnSyslogMaxTableSize (RO):

The maximum number of syslog message the syslog table can hold.

10.1.10 pdnSyslogClearTable (RW):

Valid values are noOp(1) and clear (2). This object clears all the entries in the syslog message table.

10.1.11 pdnSyslogMsgToConsole (RW):

Valid values are disable (1), enable (2). 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.

10.1.12 pdnSyslogRateLimiting (RW):

Valid values are disable (1), enable (2). This allows one to enable or disable rate limitiming. When rate limiting is enabled, it prevents the exact same message from being resent within a fixed amount of time.

10.2 Paradyne Health and Status MIB (mpe_HealthAndStatus.mib)

The Paradyne health & Status MIB is used to store the results of the Power On Self Test (POST). It is indexed by the entPhysicalIndex of the unit being selftested.

Table 1-97. Health and Status MIB

Table	Comments
mpeDevHealthAndStatusEntry	mpeDevSelfTestResults

10.2.1 mpeDevSelfTestResults (RO) :

The format for the selfTest results is as follows:

“RAM=%c, FPGAs=%c, PLDS=%c, NAND Flash=%c, File System=%c, ATM Switch=%c, SAR=%c, I2C Bus=%c, Temperature Monitor=%c, PLLs=%c, SEEP=%c, Fan Tachometer=%c, Ethernet Interface=%c, Console Interface=%c”

Where %c is:

‘P’ = Passed.

‘F’ = Failed.

‘D’ = Disabled.

‘N’ = Not available.

10.3 Traps.

10.3.1 pdnDevConfigTrapsEnable (RW) (pdnTrapMgr.mib):

This enterprise object allows NMS to turn on/off the generation of notifications in the device. The possible values are:

- enabled (1) - default
- disabled(2)

10.3.2 devLastTrapString (pdn_HealthAndStatus.mib)

This is the value of the last trap string variable appended to every supported trap.

Table 1-98. devLastTrapString (1 of 4)

Trap	devLastTrapString
GENERIC Traps	
LinkDown	"Link [*administratively*] down"
LinkUp	"Link up"
WarmStart	"Unit Reset."
AuthenticationFailureTrap	"Too many successive login errors from User at "+*IpAddress*

Table 1-98. devLastTrapString (2 of 4)

Trap	devLastTrapString
DSX Traps	
dsx1LineStatusChange	"The dsx1LineStatus has changed on a dsx1 interface."
ATM-M4 Traps	
atmfM4IfAisAlarm	"AIS alarm detected."
atmfM4IfLcdAlarm	"Loss of Cell delineation detected."
atmfM4IfLosAlarm	"Loss of Signal detected."
atmfM4IfLofAlarm	"Loss of Frame detected."
atmfM4VclXConnCreated	"VCL cross-connection has just been created."
atmfM4VclXConnDeleted	"VCL cross-connection has just been deleted."
atmfM4VclXConnChanged	"VCL cross-connection has just been changed."
atmfM4HwFanFailAlarm	"Fan alarm - [*up/down*]."
Entity MIB Traps:	
entConfigChange	"Entity configuration changed."
SHDSL MIB Traps:	
hdsl2ShdslLoopAttenCrossing	" Loop attenuation threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslSNRMarginCrossing	" SNR margin threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslPerfESThresh	"Errored Seconds (ES) threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslPerfSESThresh	"Severely Errored Seconds (SES) threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslPerfCRCanomaliesThresh	"CRC anomalies threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslPerfLOSWSThresh	"LOSW seconds threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslPerfUASThresh	"Unavailable seconds threshold has been reached/exceeded for the SHDSL segment endpoint."
hdsl2ShdslSpanInvalidNumRepeaters	"Mismatch has been detected between the number of repeater/regenerator units configured for a SHDSL line and the actual number of repeater/.regenerator discovered via the EOC."
hdsl2ShdslLoopbackFailure	"Endpoint loopback command failed for an SHDSL segment."
hdsl2ShdslpowerBackoff	"Power Backoff status has changed for this Endpoint."
hdsl2ShdsldeviceFault	"Device Failure status has changed for this Endpoint."
hhdsl2ShdsldcContinuityFault	"dc Continuity Fault status has changed for this Endpoint."
hdsl2ShdslconfigInitFailure	"Config Init Failure status has changed for this Endpoint."
hdsl2ShdslprotocolInitFailure	"Protocol Init Failure status has changed for this Endpoint."

Table 1-98. devLastTrapString (3 of 4)

Trap	devLastTrapString
hdsl2ShdslnoNeighborPresent	"No Neighbor Present status has changed for this Endpoint."
hdsl2ShdslLocalPowerLoss	" Impending unit failure due to loss of local power condition has been detected."
SONET LINEAR APS MIB Traps:	
apsEventSwitchover	"Switchover status has changed."
apsEventModeMismatch	"Mode status mismatch."
apsEventChannelMismatch	"Channel status mismatch."
apsEventPSBF	"A Protection Switch Byte Failure (PSBF) has occurred."
apsEventFEPLF	"A Far-End Protection Line Failure (FEPLF) has occurred."
Enterprise Specific Traps:	
mpeSelfTestFailure	"SelfTest failed"
mpeCcn	"Device Configuration changed."
devSonetStatusChange	"Sonet status (Section, Line or Path) has changed."
pdnRedunEventNoActiveModule	"No module is in the Active state."
pdnRedunEventHwIncompatible	"Redundancy units have hardware incompatibilities."
pdnRedunEventStandbyAlarmOrReset	"Redundancy unit *entPhysicalIndex* is in the Standby/Alarm state."
pdnRedunEventFwIncompatible	"Redundancy units have firmware incompatibilities."
pdnRedunEventCfgIncompatible	"Redundancy units have configuration incompatibilites."
ADSL Specific Traps:	
adslAtucPerfLofsThreshTrap	"(Central) Loss of Framing 15-min interval threshold reached."
adslAtucPerfLossThreshTrap	"(Central) Loss of Signal 15-min interval threshold reached."
adslAtucPerfESsThreshTrap	"(Central) Errored Second 15-min interval threshold reached."
adslAtucRateChangeTrap	"(Central) The ATUCs transmit rate has changed".
adslAtucPerfLolsThreshTrap	"(Central) Loss of Link 15-minute interval threshold reached."
adslAtucInitFailureTrap	"ATUC initialization failed on ATM port."
adslAturPerfLofsThreshTrap	"(Remote) Loss of Framing 15-minute interval threshold reached."
adslAturPerfLossThreshTrap	"(Remote) Loss of Signal 15-minute interval threshold reached."
adslAturPerfLprsThreshTrap	"(Remote) Loss of Power 15-minute interval threshold reached."
adslAturPerfESsThreshTrap	"(Remote) Errored Second 15-minute interval threshold reached."
adslAturRateChangeTrap	"(Remote) The ATURs transmit rate has changed."
adslAtucSesLThreshTrap	(Central) SES-line 15-minute threshold reached on ATM port."
adslAtucUasLThreshTrap	"(Central) UAS-line 15-minute threshold reached on ATM port."
adslAturSesLThreshTrap	"(Remote) SES-line 15-minute threshold reached on ATM port."
adslAturUasLThreshTrap	"(Remote) UAS-line 15-minute threshold reached on ATM port."

Table 1-98. devLastTrapString (4 of 4)

Trap	devLastTrapString
IMA Traps	
imaFailureAlarm	"Alarm condition on Ima interface."

Index

H

I

L

M

P

R

S

T

U

V

W