SNMP

This chapter explains Simple Network Management Protocol (SNMP) as implemented by the Cisco ONS 15600.

For SNMP setup information, refer to the *Cisco ONS 15600 Procedure Guide*.

Chapter topics include:

6.1 SNMP Overview

SNMP is an application-layer communication protocol that allows ONS 15600 network devices to exchange management information among these systems and with other devices outside the network. Through SNMP, network administrators can manage network performance, find and solve network problems, and plan network growth.

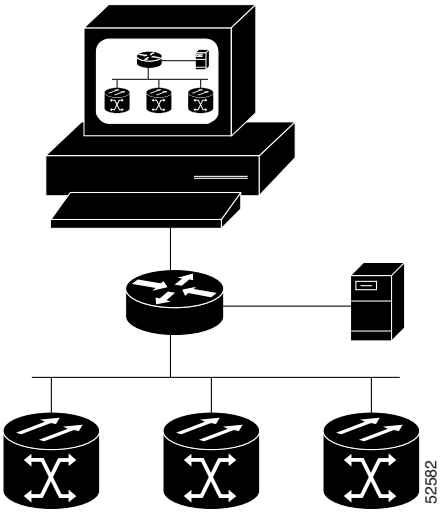
The ONS 15600 uses SNMP for asynchronous event notification to a network management system (NMS). ONS SNMP implementation uses standard Internet Engineering Task Force (IETF) management information bases (MIBs) to convey node-level inventory, fault, and performance management information for generic read-only management of SONET technologies. SNMP allows a generic SNMP manager such as HP OpenView Network Node Manager (NNM) or Open Systems Interconnection (OSI) NetExpert to be utilized for limited management functions.

The Cisco ONS 15600 supports SNMP Version 1 (SNMPv1) and SNMP Version 2c (SNMPv2c). These versions share many features, but SNMPv2c includes additional protocol operations and 64-bit performance monitoring support. This chapter describes both versions and gives SNMP configuration parameters for the ONS 15600.

Note The CERENT-MSDWDM-MIB.mib, CERENT-FC-MIB.mib, and CERENT-GENERIC-PM-MIB.mib in the CiscoV2 directory support 64-bit performance monitoring counters. The SNMPv1 MIB in the CiscoV1 directory does not contain 64-bit performance monitoring counters, but supports the lower and higher word values of the corresponding 64-bit counter. The other MIB files in the CiscoV1 and CiscoV2 directories are identical in content and differ only in format.

Figure 6-1 illustrates the basic layout idea of an SNMP-managed network.

Figure 6-1 Basic Network Managed by SNMP

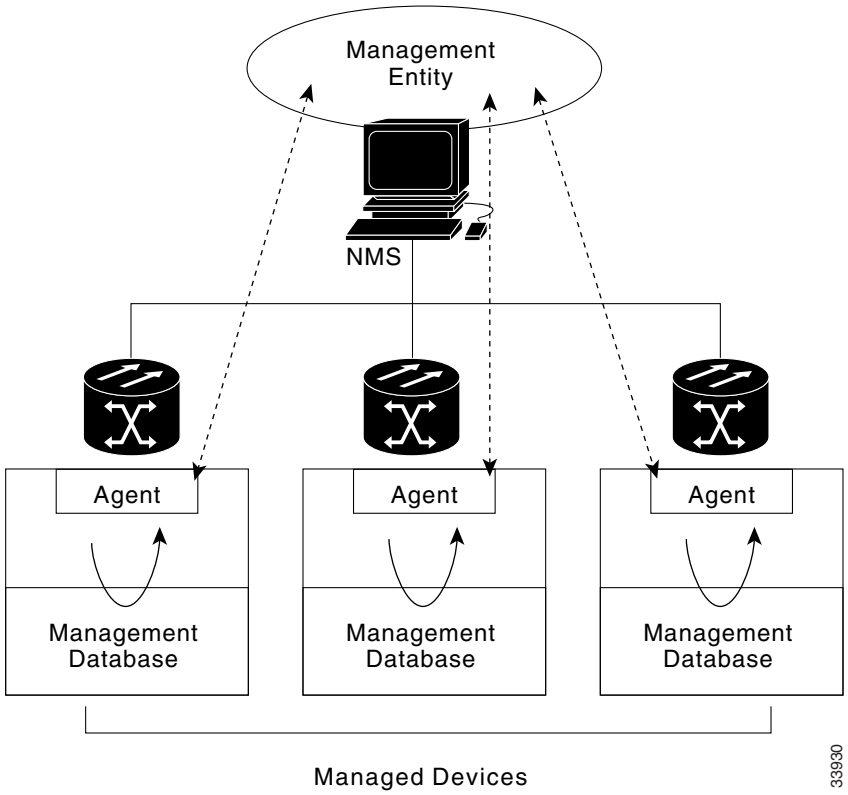


6.2 Basic SNMP Components

In general terms, an SNMP-managed network consists of a management system, agents, and managed devices.

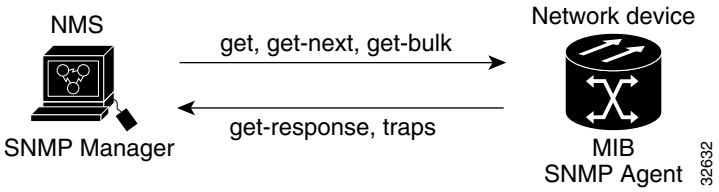
A management system such as HP OpenView executes monitoring applications and controls managed devices. Management systems execute most of the management processes and provide the bulk of memory resources used for network management. A network might be managed by one or more management systems. Figure 6-2 illustrates the relationship between the network manager, SNMP agent, and the managed devices.

Figure 6-2 Example of the Primary SNMP Components



An agent (such as SNMP) residing on each managed device translates local management information data, such as performance information or event and error information caught in software traps, into a readable form for the management system. Figure 6-3 illustrates SNMP agent get-requests that transport data to the network management software.

Figure 6-3 Agent Gathering Data from a MIB and Sending Traps to the Manager



The SNMP agent captures data from management information bases, or MIBs, which are device parameter and network data repositories, or from error or change traps.

A managed element—such as a router, access server, switch, bridge, hub, computer host, or network element (such as an ONS 15600)—is accessed through the SNMP agent. Managed devices collect and store management information, making it available through SNMP to other management systems having the same protocol compatibility.

6.3 SNMP External Interface Requirement

Since all SNMP requests come from a third-party application, the only external interface requirement is that a third-part SNMP client application can upload RFC 3273 SNMP MIB variables in the etherStatsHighCapacityTable, etherHistoryHighCapacityTable, or mediaIndependentTable.

6.4 SNMP Version Support

The ONS 15600 supports SNMPv1 and SNMPv2c traps and get requests. The ONS 15600 SNMP MIBs define alarms, traps, and status. Through SNMP, NMS applications can query a management agent for data from functional entities such as Ethernet switches and SONET multiplexers using a supported MIB.

Note ONS 15600 MIB files in the CiscoV1 and CiscoV2 directories are almost identical in content except for the difference in 64-bit performance monitoring features. The CiscoV2 directory contains three MIBs with 64-bit performance monitoring counters: CERENT-MSDWDM-MIB.mib, CERENT-FC-MIB.mib, and CERENT-GENERIC-PM-MIB.mib. The CiscoV1 directory does not contain any 64-bit counters, but it does support the lower and higher word values used in 64-bit counters. The two directories also have somewhat different formats.

6.5 SNMP Message Types

The ONS 15600 SNMP agent communicates with an SNMP management application using SNMP messages. Table 6-1 describes these messages.

Table 6-1 ONS 15600 SNMP Message Types

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves the value following the named variable; this operation is often used to retrieve variables from within a table. With this operation, an SNMP manager does not need to know the exact variable name. The SNMP manager searches sequentially to find the needed variable from within the MIB.
get-response	Replies to a get-request, get-next-request, get-bulk-request, or set-request sent by an NMS.
get-bulk-request	Fills the get-response with up to the max-repetition number of get-next interactions, similar to a get-next-request.
set-request	Provides remote network monitoring (RMON) MIB.
trap	Indicates that an event has occurred. An unsolicited message is sent by an SNMP agent to an SNMP manager.

6.6 SNMP Management Information Bases

This section contains the following information:

- 6.6.1 IETF-Standard MIBs for ONS 15600, page 6-5 lists IETF-standard MIBs that are implemented in the ONS 15600 and shows their compilation order.
- 6.6.2 Proprietary ONS 15600 MIBs, page 6-6 lists proprietary MIBs for the ONS 15600 and shows their compilation order.

6.6.1 IETF-Standard MIBs for ONS 15600

Table 6-2 lists the IETF-standard MIBs implemented in the ONS 15600 SNMP agents.

First compile the MIBs in Table 6-2. Next, compile the MIBs in the order given in Table 6-3.

Caution If you do not compile MIBs in the correct order, one or more might not compile correctly.

Table 6-2 IETF Standard MIBs Implemented in the ONS 15600 System

RFC[1] Number	Module Name	Title/Comments
—	IANAifType-MIB.mib	Internet Assigned Numbers Authority (IANA) ifType
1213	RFC1213-MIB-rfc1213.mib	Management Information Base for Network
1907	SNMPV2-MIB-rfc1907.mib	Management of TCP/IP-based Internet: MIB-II Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)
1253	RFC1253-MIB-rfc1253.mib	OSPF Version 2 Management Information Base
1493	BRIDGE-MIB-rfc1493.mib	Definitions of Managed Objects for Bridges (This defines MIB objects for managing MAC bridges based on the IEEE 802.1D-1990 standard between Local Area Network [LAN] segments.)
2819	RMON-MIB-rfc2819.mib	Remote Network Monitoring Management Information Base
2737	ENTITY-MIB-rfc2737.mib	Entity MIB (Version 2)
2233	IF-MIB-rfc2233.mib	Interfaces Group MIB using SNMPv2
2358	EtherLike-MIB-rfc2358.mib	Definitions of Managed Objects for the Ethernet-like Interface Types
2493	PerfHist-TC-MIB-rfc2493.mib	(Not applicable to the ONS 15600) Textual Conventions for MIB Modules Using Performance History Based on 15 Minute Intervals
2495	DS1-MIB-rfc2495.mib	Not applicable to the ONS 15600
2496	DS3-MIB-rfc2496.mib	Not applicable to the ONS 15600
2558	SONET-MIB-rfc2558.mib	Definitions of Managed Objects for the SONET/SDH Interface Type

Table 6-2 IETF Standard MIBs Implemented in the ONS 15600 System (continued)

RFC[1] Number	Module Name	Title/Comments
2674	P-BRIDGE-MIB-rfc2674.mib Q-BRIDGE-MIB-rfc2674.mib	Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions
3273	HC-RMON-MIB	The MIB module for managing RMON device implementations, augmenting the original RMON MIB as specified in RFC 2819 and RFC 1513, and RMON-2 MIB as specified in RFC 2021

1. RFC = Request for Comment

6.6.2 Proprietary ONS 15600 MIBs

Each ONS 15600 is shipped with a software CD containing applicable proprietary MIBs. The MIBs in Table 6-3 lists the proprietary MIBs for the ONS 15600.

Table 6-3 ONS 15600 Proprietary MIBs

MIB Number	Module Name
1	CERENT-GLOBAL-REGISTRY.mib
2	CERENT-TC.mib
3	CERENT-600.mib
4	CERENT-GENERIC.mib

Note If you cannot compile the proprietary MIBs correctly, log into the Cisco Technical Assistance Center (TAC) at http://www.cisco.com/techsupport or call Cisco TAC (800) 553-2447.

6.7 SNMP Trap Content

The ONS 15600 generates all alarms and events, such as raises and clears, as SNMP traps. These contain the following information:

- Object IDs that uniquely identify each event with information about the generating entity (the slot or port; synchronous transport signal [STS] and Virtual Tributary [VT], or bidirectional line switched ring [BLSR]).
- Severity and service effect of the alarm (Critical [CR], Major [MJ], Minor [MN], or event; Service-Affecting [SA] or Non Service Affecting [NSA]).
- Date and time stamp showing when the alarm occurred.

6.7.1 Generic and IETF Traps

Table 6-4 contains information about the generic threshold and performance monitoring MIBs that can be used to monitor any network element (NE) contained in the network. The ONS 15600 supports the generic IETF traps listed in Table 6-4.

Table 6-4 ONS 15600 Generic Traps

Trap	From RFC No. MIB	Description
coldStart	RFC1213-MIB	Agent up, cold start.
warmStart	RFC1213-MIB	Agent up, warm start.
entConfigChange	RFC2037/ ENTITY-MIB	The entLastChangeTime value has changed.

6.7.2 Variable Trap Bindings

Each SNMP trap contains variable bindings that are used to create the MIB tables. Variable bindings for the ONS 15600 are listed in Table 6-5. For each group (such as Group A), all traps within the group are associated with all of its variable bindings.

Table 6-5 15600 SNMPv2 Trap Variable Bindings

Group	Associated Trap Name(s)	(Variable Binding Number)	SNMPv2 Variable Bindings	Description
A	dsx1LineStatusChange (from RFC 2495, not applicable to ONS 15600 but applicable to other platforms)	(1)	dsx1LineStatus	This variable indicates the line status of the interface. It contains loopback, failure, received alarm and transmitted alarm information.
		(2)	dsx1LineStatusLastChange	The value of MIB II's sysUpTime object at the time this DS1 entered its current line status state. If the current state was entered prior to the last proxy-agent re-initialization, the value of this object is zero.
		(3)	cerentGenericNodeTime	The time that an event occurred.
		(4)	cerentGenericAlarmState	The alarm severity and service-affecting status. Severities are Minor (MN), Major (MJ), and Critical (CR). Service-affecting statuses are Service-Affecting (SA) and Non-Service Affecting (NSA).
		(5)	snmpTrapAddress	The address of the SNMP trap.

Table 6-5 15600 SNMPv2 Trap Variable Bindings (continued)

Group	Associated Trap Name(s)	(Variable Binding Number)	SNMPv2 Variable Bindings	Description
B	dsx3LineStatusChange (from RFC 2496, not applicable to ONS 15600 but applicable to other platforms)	(1)	dsx3LineStatus	This variable indicates the line status of the interface. It contains loopback state information and failure state information.
		(2)	dsx3LineStatusLastChange	The value of MIB II's sysUpTime object at the time this DS3/E3 entered its current line status state. If the current state was entered prior to the last reinitialization of the proxy-agent, then the value is zero.
		(3)	cerentGenericNodeTime	The time that an event occurred.
		(4)	cerentGenericAlarmState	The alarm severity and service-affecting status. Severities are Minor, Major, and Critical. Service-affecting statuses are Service-Affecting and Non-Service Affecting.
		(5)	snmpTrapAddress	The address of the SNMP trap.
C	coldStart (from RFC 1907)	(1)	cerentGenericNodeTime	The time that an event occurred.
	warmStart (from RFC 1907)	(2)	cerentGenericAlarmState	The alarm severity and service-affecting status. Severities are Minor (MN), Major (MJ), and Critical (CR). Service-affecting statuses are Service-Affecting (SA) and Non-Service Affecting (NSA).
	newRoot (from RFC)	(3)	snmpTrapAddress	The address of the SNMP trap (not supported for ONS 15600).
	topologyChange (from RFC)	—	—	(Not supported for ONS 15600)
	entConfigChange (from RFC 2737)	—	—	—
	authenticationFailure (from RFC 1907)	—	—	—

Table 6-5 15600 SNMPv2 Trap Variable Bindings (continued)

Group	Associated Trap Name(s)	(Variable Binding Number)	SNMPv2 Variable Bindings	Description
D	failureDetectedExternal ToTheNE (from CERENT-600-mib)	(1)	cerentGenericNodeTime	The time that an event occurred.
		(2)	cerentGenericAlarmState	The alarm severity and service-affecting status. Severities are Minor (MN), Major (MJ), and Critical (CR). Service-affecting statuses are Service-Affecting (SA) and Non-Service Affecting (NSA).
		(3)	cerentGenericAlarmObject Type	The entity that raised the alarm. The NMS should use this value to decide which table to poll for further information about the alarm.
		(4)	cerentGenericAlarmObject Index	Every alarm is raised by an object entry in a specific table. This variable is the index of objects in each table; if the alarm is interface-related, this is the index of the interface in the interface table.
		(5)	cerentGenericAlarmSlot Number	The slot of the object that raised the alarm. If a slot is not relevant to the alarm, the slot number is zero.
		(6)	cerentGenericAlarmPort Number	The port of the object that raised the alarm. If a port is not relevant to the alarm, the port number is zero.
		(7)	cerentGenericAlarmLine Number	The object line that raised the alarm. If a line is not relevant to the alarm, the line number is zero.
		(8)	cerentGenericAlarmObject Name	The TL1-style user-visible name that uniquely identifies an object in the system.
		(9)	cerentGenericAlarm AdditionalInfo	Additional information for the alarm object. In the current version of the MIB, this object contains provisioned description for alarms that are external to the NE. If there is no additional information, the value is zero.
		(10)	snmpTrapAddress	The address of the SNMP trap.

Table 6-5 15600 SNMPv2 Trap Variable Bindings (continued)

Group	Associated Trap Name(s)	(Variable Binding Number)	SNMPv2 Variable Bindings	Description
E	performanceMonitorThresholdCrossingAlert (from CERENT-600-mib)	(1)	cerentGenericNodeTime	The time that an event occurred.
		(2)	cerentGenericAlarmState	The alarm severity and service-affecting status. Severities are Minor (MN), Major (MJ), and Critical (CR). Service-affecting statuses are Service-Affecting (SA) and Non-Service Affecting (NSA).
		(3)	cerentGenericAlarmObjectType	The entity that raised the alarm. The NMS should use this value to decide which table to poll for further information about the alarm.
		(4)	cerentGenericAlarmObjectIndex	Every alarm is raised by an object entry in a specific table. This variable is the index of objects in each table; if the alarm is interface-related, this is the index of the interface in the interface table.
		(5)	cerentGenericAlarmSlotNumber	The slot of the object that raised the alarm. If a slot is not relevant to the alarm, the slot number is zero.
		(6)	cerentGenericAlarmPortNumber	The port of the object that raised the alarm. If a port is not relevant to the alarm, the port number is zero.
		(7)	cerentGenericAlarmLineNumber	The object line that raised the alarm. If a line is not relevant to the alarm, the line number is zero.
		(8)	cerentGenericAlarmObjectName	The TL1-style user-visible name that uniquely identifies an object in the system.
		(9)	cerentGenericThresholdMonitorType	This object indicates the type of metric being monitored.
		(10)	cerentGenericThresholdLocation	Indicates whether the event occurred at the near or far end.
		(11)	cerentGenericThresholdPeriod	Indicates the sampling interval period.
		(12)	cerentGenericThresholdSetValue	The value of this object is the threshold provisioned by the NMS.
		(13)	cerentGenericThresholdCurrentValue	—
		(14)	cerentGenericThresholdDetectType	—
		(15)	snmpTrapAddress	The address of the SNMP trap.

Table 6-5 15600 SNMPv2 Trap Variable Bindings (continued)

Group	Associated Trap Name(s)	(Variable Binding Number)	SNMPv2 Variable Bindings	Description
F	All other traps (from CERENT-600-MIB) not listed above	(1)	cerentGenericNodeTime	The time that an event occurred.
		(2)	cerentGenericAlarmState	The alarm severity and service-affecting status. Severities are Minor (MN), Major (MJ), and Critical (CR). Service-affecting statuses are Service-Affecting (SA) and Non-Service Affecting (NSA).
		(3)	cerentGenericAlarmObject Type	The entity that raised the alarm. The NMS should use this value to decide which table to poll for further information about the alarm.
		(4)	cerentGenericAlarmObject Index	Every alarm is raised by an object entry in a specific table. This variable is the index of objects in each table; if the alarm is interface-related, this is the index of the interface in the interface table.
		(5)	cerentGenericAalrmSlot Number	The slot of the object that raised the alarm. If a slot is not relevant to the alarm, the slot number is zero.
		(6)	cerentGenericAlarmPort Number	The port of the object that raised the alarm. If a port is not relevant to the alarm, the port number is zero.
		(7)	cerentGenericAlarmLine Number	The object line that raised the alarm. If a line is not relevant to the alarm, the line number is zero.
		(8)	cerentGenericAlarmObject Name	The TL1-style user-visible name that uniquely identifies an object in the system.
		(9)	snmpTrapAddress	The address of the SNMP trap.

6.8 Proxy Over Firewalls

SNMP and NMS applications have traditionally been unable to cross firewalls used for isolating security risks inside or from outside networks. Release 6.0 CTC enables network operations centers (NOCs) to access performance monitoring data such as RMON statistics or autonomous messages across firewalls by using an SMP proxy element installed on a firewall.

The application-level proxy transports SNMP protocol data units (PDU) between the NMS and NEs, allowing requests and responses between the NMS and NEs and forwarding NE autonomous messages to the NMS. The proxy agent requires little provisioning at the NOC and no additional provisioning at the NEs.

The firewall proxy is intended for use in a gateway network element-end network element (GNE-ENE) topology with many NEs through a single NE gateway. Up to 64 SNMP requests (such as get, getnext, or getbulk) are supported at any time behind single or multiple firewalls. The proxy interoperates with common NMS such as HP OpenView.

For security reasons, the SNMP proxy feature must be enabled at all receiving and transmitting NEs to function. For instructions to do this, refer to the *Cisco ONS 15600 Procedure Guide*.

6.8.1 Remote Monitoring

The ONS 15600 incorporates RMON to allow network operators to monitor Ethernet facility performance and events. Release 6.0 provides remote data communications channel (DCC) monitoring using 64-bit RMON over the DCC to gather historical and statistical Ethernet data. In general, ONS 15600 system RMON is based on the IETF-standard MIB RFC 2819 and includes the following five groups from the standard MIB: Ethernet Statistics, History Control, Ethernet History, Alarm, and Event.

6.8.2 64-Bit RMON Monitoring over DCC

The ONS 15600 DCC is implemented over the IP protocol, which is not compatible with Ethernet. The system monitors Ethernet equipment history and statistics using RMON. This release adds RMON DCC monitoring (for both IP and Ethernet) to monitor the health of remote DCC connections.

In R6.0, the implementation contains two MIBs for DCC interfaces. They are:

- cMediaIndependentTable—Standard, RFC 3273; the proprietary extension of the HC-RMON MIB used for reporting statistics
- cMediaIndependentHistoryTable—Proprietary MIB used to support history

6.8.2.1 Row Creation in MediaIndependentTable

The mediaIndependentTable is created automatically when the Ethernet facility is created on the ONS 15600 ASAP card.

6.8.2.2 Row Creation in cMediaIndependentHistoryControlTable

SNMP row creation and deletion for the cMediaIndependentHistoryControlTable follows the same processes as for the MediaIndependentTable; only the variables differ.

In order to create a row, the SetRequest PDU should contain the following:

- cMediaIndependentHistoryControlDataSource and its desired value
- cMediaIndependentHistoryControlOwner and its desired value
- cMediaIndependentHistoryControlStatus with a value of createRequest (2)

6.8.3 HC-RMON-MIB Support

For the ONS 15600, the implementation of the high-capacity remote monitoring information base (HC-RMON-MIB, or RFC 3273) enables 64-bit support of existing RMON tables. This support is provided with the etherStatsHighCapacityTable and the etherHistoryHighCapacityTable. An additional table, the mediaIndependentTable, and an additional object, hcRMONCapabilities, are also added for this support. All of these elements are accessible by any third-party SNMP client having RFC 3273 support.

6.8.4 Ethernet Statistics RMON Group

The Ethernet Statistics group contains the basic statistics monitored for each subnetwork in a single table called the etherStatsTable.

6.8.4.1 Row Creation in etherStatsTable

The SetRequest PDU for creating a row in this table should contain all the values needed to activate a row in a single set operation, and an assigned status variable to createRequest. The SetRequest PDU object ID (OID) entries must all carry an instance value, or type OID, of 0.

In order to create a row, the SetRequest PDU should contain the following:

- The etherStatsDataSource and its desired value
- The etherStatsOwner and its desired value (size of this value is limited to 32 characters)
- The etherStatsStatus with a value of createRequest (2)

The etherStatsTable creates a row if the SetRequest PDU is valid according to the above rules. When the row is created, the SNMP agent decides the value of etherStatsIndex. This value is not sequentially allotted or contiguously numbered. It changes when an Ethernet interface is added or deleted. The newly created row will have etherStatsStatus value of valid (1).

If the etherStatsTable row already exists, or if the SetRequest PDU values are insufficient or do not make sense, the SNMP agent returns an error code.

Note EtherStatsTable entries are not preserved if the SNMP agent is restarted.

6.8.4.2 Get Requests and GetNext Requests

Get requests and getNext requests for the etherStatsMulticastPkts and etherStatsBroadcastPkts columns return a value of zero because the variables are not supported by ONS 15600 Ethernet facilities.

6.8.4.3 Row Deletion in etherStatsTable

To delete a row in the etherStatsTable, the SetRequest PDU should contain an etherStatsStatus value of 4 (invalid). The OID marks the row for deletion. If required, a deleted row can be recreated.

6.8.5 History Control RMON Group

The History Control group defines sampling functions for one or more monitor interfaces in the historyControlTable. The values in this table, as specified in RFC 2819, are derived from the historyControlTable and etherHistoryTable.

6.8.5.1 History Control Table

The RMON is sampled at one of four possible intervals. Each interval, or period, contains specific history values (also called buckets). Table 6-6 lists the four sampling periods and corresponding buckets.

The historyControlTable maximum row size is determined by multiplying the number of ports on a card by the number of sampling periods. For example, an ONS 15600 E100 card contains 24 ports, which multiplied by periods allows 96 rows in the table. An E1000 card contains 14 ports, which multiplied by four periods allows 56 table rows.

Table 6-6 RMON History Control Periods and History Categories

Sampling Periods (historyControlValue Variable)	Total Values, or Buckets (historyControl Variable)
15 minutes	32
24 hours	7
1 minute	60
60 minutes	24

6.8.5.2 Row Creation in historyControlTable

The etherStats table and historyControl table are automatically created when the Ethernet facility is created. History size is based upon the default history bucket located in Table 6-6.

6.8.5.3 Get Requests and GetNext Requests

These PDUs are not restricted.

6.8.5.4 Row Deletion in historyControl Table

To delete a row from the table, the SetRequest PDU should contain a historyControlStatus value of 4 (invalid). A deleted row can be recreated.

6.8.5.5 Ethernet History RMON Group

The ONS 15600 implements the etherHistoryTable as defined in RFC 2819. The group is created within the bounds of the historyControlTable and does not deviate from the RFC in its design.

6.8.5.6 64-Bit etherHistoryHighCapacityTable

64-bit Ethernet history for the HC-RMON-MIB is implemented in the etherHistoryHighCapacityTable, which is an extension of the etherHistoryTable. The etherHistoryHighCapacityTable adds four columns for 64-bit performance monitoring data. These two tables have a one-to-one relationship. Adding or deleting a row in one table will effect the same change in the other.

6.8.5.7 Alarm RMON Group

The Alarm group consists of the alarmTable, which periodically compares sampled values with configured thresholds and raises an event if a threshold is crossed. This group requires the implementation of the event group, which follows this section.

6.8.5.8 Alarm Table

The NMS uses the alarmTable to determine and provision network performance alarmable thresholds.

6.8.5.9 Get Requests and GetNext Requests

These PDUs are not restricted.

6.8.5.10 Row Deletion in alarmTable

To delete a row from the table, the SetRequest PDU should contain an alarmStatus value of 4 (invalid). A deleted row can be recreated. Entries in this table are preserved if the SNMP agent is restarted.

6.8.5.11 Event RMON Group

The Event group controls event generation and notification. It consists of two tables: the eventTable, which is a read-only list of events to be generated, and the logTable, which is a writable set of data describing a logged event. The ONS 15600 implements the logTable as specified in RFC 2819.

6.8.5.12 Event Table

The eventTable is read-only and unprovisionable. The table contains one row for rising alarms and another for falling ones. This table has the following restrictions:

- The eventType is always “log-and-trap (4)”.
- The eventCommunity value is always a zero-length string, indicating that this event causes the trap to be despatched to all provisioned destinations.
- The eventOwner column value is always “monitor.”
- The eventStatus column value is always “valid(1)”.

6.8.5.13 Log Table

The logTable is implemented exactly as specified in RFC 2819. The logTable is based upon data that is locally cached in a controller card. If there is a controller card protection switch, the existing logTable is cleared and a new one is started on the newly active controller card. The table contains as many rows as provided by the alarm controller.