Avaya Virtual Services Platform 7000 Series Configuration System Monitoring

Release 10.1
NN47202-505, 02.02
March 2012

© 2012 Avaya Inc.

All Rights Reserved.

Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer

“Documentation” means information published by Avaya in varying mediums which may include product information, operating instructions and performance specifications that Avaya generally makes available to users of its products. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original published version of documentation unless such modifications, additions, or deletions were performed by Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer

Avaya is not responsible for the contents or reliability of any linked Web sites referenced within this site or documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty

Avaya provides a limited warranty on its Hardware and Software (“Product(s)”). Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya’s standard warranty language, as well as information regarding support for this Product while under warranty is available to Avaya customers and other parties through the Avaya Support Web site: http://support.avaya.com. Please note that if you acquired the Product(s) from an authorized Avaya reseller outside of the United States and Canada, the warranty is provided to you by said Avaya reseller and not by Avaya.

Licenses

THE SOFTWARE LICENSE TERMS AVAILABLE ON THE AVAYA WEBSITE, HTTP://SUPPORT.AVAYA.COM/LICENSEINFO/ ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AUTHORIZED AVAYA RESELLER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AUTHORIZED AVAYA RESELLER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA AUTHORIZED RESELLER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS “YOU” AND “END USER”), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE AVAYA AFFILIATE (“AVAYA”).

Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, or Hardware provided by Avaya. All content on this site, the documentation and the Product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Third-party components

Certain software programs or portions thereof included in the Product may contain software distributed under third party agreements (“Third Party Components”), which may contain terms that expand or limit rights to use certain portions of the Product (“Third Party Terms”). Information regarding distributed Linux OS source code (for those Products that have distributed the Linux OS source code), and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply to them is available on the Avaya Support Web site: http://support.avaya.com/Copyright.

Trademarks

The trademarks, logos and service marks (“Marks”) displayed in this site, the Documentation and Product(s) provided by Avaya are the registered or unregistered Marks of Avaya, its affiliates, or other third parties. Users are not permitted to use such Marks without prior written consent from Avaya or such third party which may own the Mark. Nothing contained in this site, the Documentation and Product(s) should be construed as granting, by implication, estoppel, or otherwise, any license or right in and to the Marks without the express written permission of Avaya or the applicable third party.

Avaya is a registered trademark of Avaya Inc.

All non-Avaya trademarks are the property of their respective owners, and “Linux” is a registered trademark of Linus Torvalds.

Downloading Documentation

For the most current versions of Documentation, see the Avaya Support Web site: http://support.avaya.com.

Contact Avaya Support

Avaya provides a telephone number for you to use to report problems or to ask questions about your Product. The support telephone number is 1-800-242-2121 in the United States. For additional support telephone numbers, see the Avaya Web site: http://support.avaya.com.

Contents

Chapter 1: Purpose of this document 7
Chapter 2: New in this release 9
Features 9
ACLI audit log 9
Enterprise Device Manager (EDM) 9
RMON Ethernet history and statistics 9
Stack Loopback tests 10
Stack Health Check 10
Stack Health Monitoring and Recovery 10
Stack Monitor 11
Stack Port Counters 11
Chapter 3: System monitoring fundamentals 13
System logging 13
ACLI audit log 13
Remote logging 14
Dual syslog server support 14
Alarms 15
How RMON alarms work 15
Alarm variables and types 16
Port mirroring 17
Port-based mirroring configuration 17
Address-based mirroring configuration 18
CPU and memory utilization 19
Remote Network Monitoring 20
Environmental information 20
Stack loopback tests 20
Stack Health Check 21
Stack Health Monitoring and Recovery 21
Stack Monitor 22
Chapter 4: System diagnostics and statistics using ACLI 25
Viewing port statistics using ACLI 25
Displaying port operational status using ACLI 28
Validating port operational status using ACLI 29
Displaying port information using ACLI 29
Viewing environmental information using ACLI 30
Enabling the ACLI audit log using ACLI 31
Disabling the ACLI audit log using ACLI 31
Displaying the ACLI audit log using ACLI 32
Displaying Stack Health Check using ACLI 32
Configuring the Stack Monitor using ACLI 34
Configuring Stack Health Monitoring and Recovery using ACLI 35
Viewing Stack Port Counters using ACLI 36
Testing Stack Loopback using ACLI 37
Chapter 5: Network monitoring configuration using ACLI 41

Viewing CPU utilization using ACLI 41
Viewing memory utilization using ACLI 41
Configuring the system log using ACLI 41
Displaying the system log using ACLI 42
Disabling the system log using ACLI 43
Setting the system log to default using ACLI 43
Clearing the system log using ACLI 43
Configuring remote system logging using ACLI 44
Disabling remote system logging using ACLI 46
Restoring remote system logging to default using ACLI 47
Performing a traceroute using ACLI 47
Displaying IPv6 interface statistics using ACLI 49
Displaying IPv6 interface ICMP statistics using ACLI 49
Displaying IPv6 UDP statistics using ACLI 49
Displaying IPv6 UDP endpoints using ACLI 50
Displaying IPv6 TCP statistics using ACLI 50
Displaying IPv6 TCP connections using ACLI 50
Displaying IPv6 TCP listeners using ACLI 51
Chapter 6: RMON configuration using ACLI 53
Viewing RMON alarms using ACLI 53
Viewing RMON events using ACLI 53
Viewing RMON ethernet history using ACLI 53
Viewing RMON ethernet statistics using ACLI 54
Viewing RMON history using ACLI 55
Viewing RMON statistics using ACLI 55
Setting RMON alarms using ACLI 56
Deleting RMON alarm table entries using ACLI 57
Configuring RMON event log and traps using ACLI 57
Deleting RMON event table entries using ACLI 58
Configuring RMON history using ACLI 59
Deleting RMON history table entries using ACLI 59
Configuring RMON statistics using ACLI 60
Disabling RMON statistics using ACLI 61
Chapter 7: System diagnostics and statistics using Enterprise Device Manager 63
Displaying switch summary in EDM 63
Displaying switch temperature using EDM 64
Displaying switch fan information using EDM 64
Displaying power supply information using EDM 65
Displaying stack health using EDM 66
Displaying stack information using EDM 66
Editing switch unit location in the stack using EDM 68
Configuring Stack Monitor using EDM 70
Creating a graph using EDM 70
Graphing chassis IP statistics using EDM 71
Graphing chassis ICMP In Statistics using EDM 73
Graphing chassis ICMP Out statistics using EDM 75
Graphing chassis TCP statistics using EDM 76

Graphing chassis UDP statistics using EDM ... 78
Graphing chassis LACP diagnostic statistics using EDM ... 79
Graphing chassis SNMP statistics using EDM ... 80
Graphing chassis bridge statistics using EDM ... 83
Displaying unit information using EDM ... 84
Managing unit statistics using EDM ... 84
Displaying unit statistics using EDM ... 84
Graphing unit statistics using EDM ... 85
Graphing port interface statistics using EDM ... 86
Graphing port Ethernet error statistics using EDM ... 88
Graphing port RMON statistics using EDM ... 91
Graphing miscellaneous port statistics using EDM ... 94
Chapter 8: Network monitoring configuration using Enterprise Device Manager ... 95
Displaying CPU use with EDM ... 95
Displaying memory use with EDM ... 96
Displaying system logs using EDM ... 97
Configuring the system log using EDM ... 98
Configuring remote system logging using EDM ... 99
Displaying port mirroring using EDM ... 100
Configuring port mirroring using EDM ... 102
Displaying pluggable ports using EDM ... 103
Displaying TCP information for the switch using EDM ... 104
Displaying TCP connections using EDM ... 105
Displaying TCP Listeners using EDM ... 106
Displaying UDP endpoints using EDM ... 107
Graphing IPv6 interface ICMP statistics using EDM ... 109
Displaying ICMP message statistics using EDM ... 110
Displaying IPv6 TCP global properties using EDM ... 110
Displaying IPv6 TCP connections using EDM ... 111
Displaying IPv6 TCP listeners using EDM ... 112
Displaying IPv6 UDP endpoints using EDM ... 113
Displaying network topology information using EDM ... 114
Displaying the topology table using EDM ... 115
Using the EDM MIB Web page for SNMP Get and Get-Next ... 116
Using the EDM MIB Web page for SNMP walk ... 117
Chapter 9: RMON configuration using Enterprise Device Manager ... 119
Managing RMON alarms using EDM ... 119
Displaying RMON alarms using EDM ... 119
Creating a RMON alarm using EDM ... 121
Configuring RMON alarms using EDM ... 123
Deleting a RMON alarm using EDM ... 125
Managing log information using EDM ... 128
Managing RMON events using EDM ... 129
Displaying RMON events using EDM ... 129
Creating a RMON event using EDM ... 130
Deleting a RMON event using EDM ... 132
Managing RMON history using EDM ... 132

Displaying RMON history using EDM 132
Creating a history using EDM 134
Deleting RMON history table entries using EDM 135
Managing RMON statistics using EDM 136
Displaying RMON statistics using EDM 137
Enabling ethernet statistics gathering using EDM 139
Disabling RMON ethernet statistics gathering using EDM 140

Chapter 1: Purpose of this document

This document provides system monitoring concepts and procedures for the Avaya Virtual Services Platform (VSP) 7000 Series.

This document provides procedures for both the Avaya Command Line Interface (ACLI) and Enterprise Device Manager (EDM) for system monitoring tasks. You can access ACLI through either a direct console connection to the switch or by using the Telnet or SSH protocols to connect to the switch remotely. You can access EDM through a web browser to connect to the switch remotely.

The topics discussed in this document are provided with the following assumptions:

- You have a basic knowledge of networks, Ethernet bridging, and IP routing.
- You are familiar with networking concepts and terminology.
- You have basic knowledge of network topologies.
- You have experience with Graphical User Interface (GUI).
- You have experience with the following ACLI command modes:
    - User Executive
    - Privileged EXEC
    - Global configuration
    - Interface configuration

For detailed information about the ACLI command modes, see *Avaya Virtual Services Platform 7000 Series Fundamentals* (NN47202-101).

Comments? infodev@avaya.com

Chapter 2: New in this release

The following sections indicate what's new in this document for Avaya Virtual Services Platform 7000 (VSP 7000) Series Release 10.1.

Features

See the following sections for information about feature changes.

ACLI audit log

ACLI audit log provides tracking for all ACLI commands.

A read-only area of NVRAM on the switch is reserved for an ACLI audit log, to store ACLI command history. When you enable the ACLI audit log, every time you issue an ACLI command, the switch generates and stores an audit message.

- ACLI audit log on page 13

Enterprise Device Manager (EDM)

Enterprise Device Manager (EDM) is an embedded element management and configuration application you can use to configure and manage the switch or stack. EDM runs on the switch and provides a Web-based graphical user interface through a standard web browser.

- System diagnostics and statistics using Enterprise Device Manager on page 63
- RMON configuration using Enterprise Device Manager on page 119
- Network monitoring configuration using Enterprise Device Manager on page 95

RMON Ethernet history and statistics

- Viewing RMON ethernet history using ACLI on page 53
- Viewing RMON ethernet statistics using ACLI on page 54

- Enabling ethernet statistics gathering using EDM on page 139
- Disabling RMON ethernet statistics gathering using EDM on page 140

Stack Loopback tests

You can test your Fabric Interconnect (FI) ports and FI cables by using the stack loopback test. The stack loopback test is useful to determine whether the source of the problem is a defective Fabric Interconnect (FI) cable or a damaged FI port. The test can help prevent unnecessarily sending switches for service. Two types of loopback tests exist. The internal loopback test verifies that the FI ports are functional. The external loopback test checks if the FI cables are functional.

- Stack loopback tests on page 20
- Testing Stack Loopback using ACLI on page 37

Stack Health Check

The Stack Health Check feature provides information on the state of each units Fabric Interconnect (FI) port. Stack Health Check runs a high-level test to monitor the FI port status for each unit, the number of units in a stack, if a stack runs with a temporary base unit, and to monitor stack continuity.

- Stack Health Check on page 21
- Displaying Stack Health Check using ACLI on page 32
- Displaying stack health using EDM on page 66

Stack Health Monitoring and Recovery

The Stack Health Monitoring and Recovery feature provides failure detection and recovery of units in a stack.

- Stack Health Monitoring and Recovery on page 21
- Configuring Stack Health Monitoring and Recovery using ACLI on page 35

Stack Monitor

The Stack Monitor can detect problems with the units in the stack and send traps.

- Stack Monitor on page 22
- Configuring the Stack Monitor using ACLI on page 34
- Configuring Stack Monitor using EDM on page 70

Stack Port Counters

The stack port counters measure packet statistics for the Fabric Interconnect ports of a stack unit. You can view or clear the stack port counter statistics using ACLI.

Important:

The stack counters measure the size of packets received and transmitted on FI-up and FI-down ports. The size of these packets is greater than the size of the packets received on front panel ports since an extra header is added to the packets to flow through the stack. The size of this header is 12 bytes, therefore another range of stack counters is incremented when sending packets with lengths close to the stack counters upper intervals limit.

The number of received and transmitted packets can be greater than the number of packets transmitted on front panel ports since there are different stack management packets transmitted and received.

- Viewing Stack Port Counters using ACLI on page 36

Chapter 3: System monitoring fundamentals

System monitoring is an important aspect of switch operation. The VSP 7000 Series provides a wide range of system monitoring options that you can use to closely monitor the operation of a switch or stack.

This chapter describes two general system monitoring aspects that you must consider when you use the VSP 7000 Series: system logging and port mirroring. Subsequent chapters provide information about specific system monitoring tools and how to use them.

System logging

The VSP 7000 Series supports system logging (syslog), a software tool to log system events for debugging and analysis.

The syslog tool can log application events. The logged events are stored in volatile RAM, nonvolatile RAM, or in a remote host.

The timestamp for logged events can be configured for UTC time by synchronizing with a SNTP server. For more information about SNTP and the system clock, see *Avaya Virtual Services Platform 7000 Series Getting Started* (NN47202–303).

ACLI audit log

ACLI audit log provides the switch with a means for tracking ACLI commands.

A read-only area of NVRAM on the switch is reserved for an ACLI audit log, to store ACLI command history. When you enable the ACLI audit log, every time you issue an ACLI command, the switch generates and stores an audit message.

Each ACLI audit log entry consists of the following information:

- timestamp
- fixed priority setting of 30 (= informational message)

- command source
  - serial console and the unit connected
  - Telnet or SSH connection and the IP address
- command status (success or failure)
- the ACLI command itself

When you enable remote logging, the switch also forwards the audit message to a remote syslog server, no matter the logging level.

You can disable the ACLI audit log to prevent the switch from writing log messages to NVRAM and from forwarding log messages to the syslog server, if a syslog server is configured.

The ACLI audit log is enabled by default.

Remote logging

Remote logging provides an enhanced level of logging by replicating system messages on a syslog server. You can collect system log messages from several switches at a central location, which eliminates the need to query each switch individually to view the log files.

You must configure the remote syslog server on the unit to log `informational`, `serious`, and `critical` messages to this remote server. The UDP packet is sent to port 514 of the configured remote syslog server.

After the IP address is in the system, you can send syslog messages to the remote syslog server. If a syslog message is generated prior to capturing the IP address of the server, the system stores up to 10 messages that are sent after the IP address of the remote server is on the system.

For information on configuring this feature, see Configuring remote system logging using ACLI on page 44.

Dual syslog server support

You can enable dual syslog server support by configuring and enabling a secondary remote syslog server to run in tandem with the first. The system then sends syslog messages simultaneously to both servers to ensure that syslog messages are logged, even if one of the servers becomes unavailable.

Alarms

Alarms are useful for identifying values of a variable that have gone out of range. Define an RMON alarm for a MIB variable that resolves to an integer value. String variables cannot be used. All alarms share the following characteristics:

- An upper and lower threshold value is defined.
- A corresponding rising and falling event occurs.
- An alarm interval or polling period is reached.

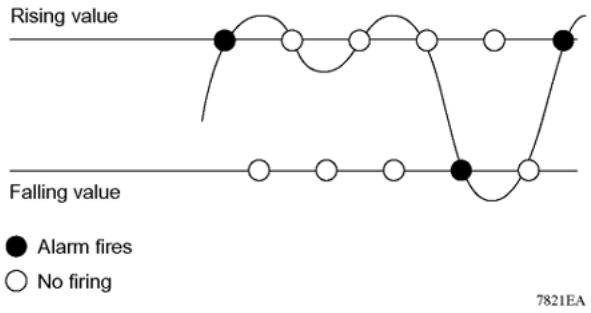
After alarms are activated, view the activity in a log or a trap log, or a script can be created to provide notification by beeping a console, sending e-mail messages, or calling a pager.

How RMON alarms work

The alarm variable is polled and the result is compared against upper and lower limit values you select after you create the alarm. If either limit is reached or crossed during the polling period, then the alarm fires and generates an event that you can view in the event log or the trap log.

The upper limit of the alarm is called the rising value, and its lower limit is called the falling value. RMON periodically samples the data based upon the alarm interval. During the first interval that the data passes above the rising value, the alarm fires as a rising event. During the first interval that the data drops below the falling value, the alarm fires as a falling event.

The following figure describes how alarms fire.



The alarm fires during the first interval that the sample goes out of range. No additional events are generated for that threshold until the opposite threshold is crossed. Therefore, it is important to carefully define the rising and falling threshold values for alarms to work as expected. Otherwise, incorrect thresholds cause an alarm to fire at every alarm interval.

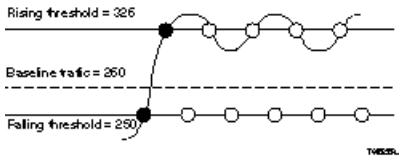
A general guideline is to define one of the threshold values to an expected baseline value, and then define the opposite threshold as the out-of-bounds limit. Because of sample averaging,

the value may be equal to ±1 of the baseline units. For example, assume an alarm is defined on octets going out of a port as the variable. The intent of the alarm is to provide notification to you after excessive traffic occurs on that port. If spanning tree is enabled, 52 octets are transmitted out of the port every 2 seconds, which is equivalent to baseline traffic of 260 octets every 10 seconds. This alarm provides notification to you if the lower limit of octets going out is defined at 260 and the upper limit is defined at 320 (or at a value greater than 260 + 52 = 312).

The first time outbound traffic other than spanning tree Bridge Protocol Data Units (BPDU) occurs, the rising alarm fires. After outbound traffic other than spanning tree ceases, the falling alarm fires. This process provides you with time intervals of a non-baseline outbound traffic.

If the alarm is defined with a falling threshold less than 260 (assuming the alarm polling interval is 10 seconds) the rising alarm can fire only once. For the rising alarm to fire a second time, the falling alarm (the opposite threshold) must fire. Unless the port becomes inactive or spanning tree is disabled (which causes the value for outbound octets to drop to zero), the falling alarm cannot fire because the baseline traffic is always greater than the value of the falling threshold. By definition, the failure of the falling alarm to fire prevents the rising alarm from firing a second time.

The following figure describes an alarm with a threshold less than 260.



Alarm variables and types

Select a variable from the variable list and a port, or other connected switch component. Some variables require port IDs, card IDs, or other indices (for example, spanning tree group IDs). Then select a rising and a falling threshold value. The rising and falling values are compared against the actual value of the variable that you choose. If the variable falls outside of the rising or falling value range, an alarm is triggered and an event is logged or trapped. For information on configuring alarms, see Setting RMON alarms using ACLI on page 56.

Configuring an alarm also involves selecting a sample type, which can be either `absolute` or `delta`. Absolute alarms are defined on the cumulative value of the alarm variable. An example of an alarm defined with absolute value is card operating status. Because this value is not cumulative, but instead represents states, such as card up (value 1) and card down (value 2), you set it for absolute value. You can create an alarm with a rising value of 2 and a falling value of 1 to alert a user to whether the card is up or down.

Most alarm variables related to Ethernet traffic are set to delta value. Delta alarms are defined based on the difference in the value of the alarm variable between the start of the polling period and the end of the polling period. Delta alarms are sampled twice for each polling period. For

each sample, the last two values are added together and compared to the threshold values. This process increases precision and allows for the detection of threshold crossings that span the sampling boundary. If you track the current values of a delta-valued alarm and add them together, therefore, the result is twice the actual value. (This result is not an error in the software.)

Port mirroring

You can designate a switch port to monitor traffic on any other specified switch ports (port-based) or to monitor traffic to or from any two specified addresses that the switch learned (address-based).

A probe device, such as the Avaya StackProbe or equivalent, must connect to the designated monitor port to use this feature. Contact an Avaya sales agent for details about the StackProbe.

Port-based mirroring configuration

The following figure shows an example of a port-based mirroring configuration in which port 20 is designated as the monitor port for ports 21 and 22 of Switch S1. Although this example shows ports 21 and 22 monitored by the monitor port (port 20), any trunk member of T1 and T2 can also be monitored.

Although this example shows port X and port Y as members of Trunk T1 and Trunk T2, they are not required to always be members of Trunk T1 and Trunk T2.

You cannot monitor trunks and you cannot configure trunk members as monitor ports.

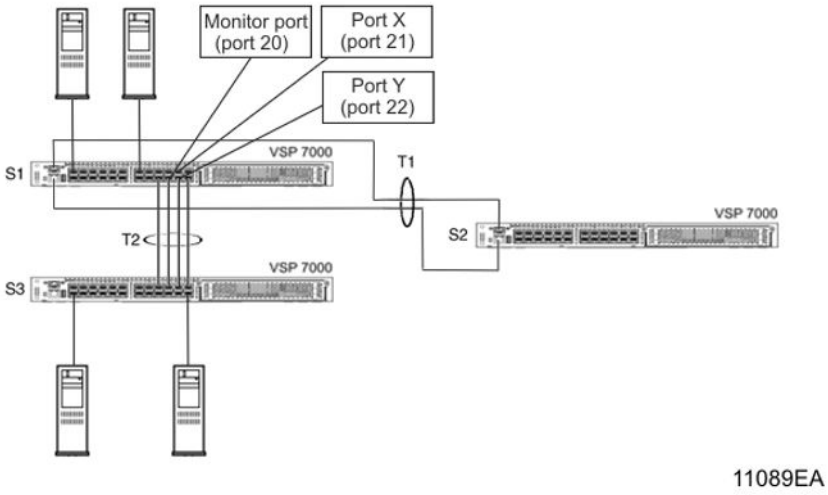


Figure 1: Port-based mirroring example

In the preceding configuration example, you can configure the designated monitor port (port 20) to monitor traffic in any of the following modes:

- Monitor all traffic received by port X.
- Monitor all traffic transmitted by port X.
- Monitor all traffic received by port X or transmitted by port Y.
- Monitor all traffic received by port X (destined to port Y) and then transmitted by port Y.
- Monitor all traffic received or transmitted by port X and transmitted or received by port Y (conversations between port X and port Y).

Address-based mirroring configuration

The following example shows an address-based mirroring configuration in which port 20, the designated monitor port for Switch S1, monitors traffic occurring between address A and address B.

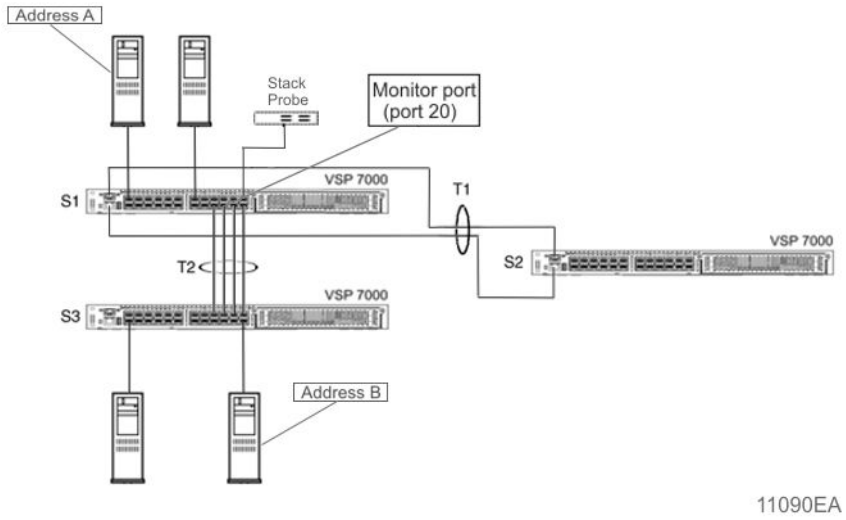


Figure 2: Address-based mirroring example

In this configuration, the designated monitor port (port 20) can be set to monitor traffic in any of the following modes:

- Monitor all traffic transmitted from address A to any address.
- Monitor all traffic received by address A from any address.
- Monitor all traffic received by or transmitted by address A.
- Monitor all traffic transmitted by address A to address B.
- Monitor all traffic between address A and address B (conversation between the two stations).

CPU and memory utilization

The CPU utilization feature provides data for CPU and memory utilization. You can view CPU utilization information for the past 10 seconds (s), 1minute (min), 1 hour (hr), 24 hr, or since system startup. The switch displays CPU utilization as a percentage. With CPU utilization information you can see the CPU use during a specific time interval.

The memory utilization feature provides information about the percentage of the dynamic memory currently used by the system. The switch displays memory utilization in terms of the lowest percentage of dynamic memory available since system startup.

No configuration is required for this display-only feature.

Remote Network Monitoring

The Remote Network Monitoring (RMON) MIB is an interface between the RMON agent on the VSP 7000 Series and an RMON management application, such as the Enterprise Device Manager.

RMON defines objects that are suitable for managing any type of network, but some groups are targeted specifically for Ethernet networks.

The RMON agent continuously collects statistics and monitors switch performance.

RMON provides three major functions:

- creating and displaying alarms for user-defined events
- gathering cumulative statistics for Ethernet interfaces
- tracking a history of statistics for Ethernet interfaces

Environmental information

Environmental information provides information on the status of the environment of each unit in a stack. You can perform the following tasks:

- Monitor the hardware status for each unit.
- Detect the presence of AC, DC, or AC/DC power.
- Monitor the CPUs temperature.
- Identify damaged or missing hardware.

Stack loopback tests

You can test your Fabric Interconnect (FI) ports and FI cables by using the stack loopback test. The stack loopback test is useful to determine whether the source of the problem is a defective FI cable or a damaged FI port. The test can help prevent unnecessarily sending switches for service.

Two types of loopback tests exist. The internal loopback test verifies that the FI ports are functional. The external loopback test checks if the FI cables are functional. Perform the

external loopback test by connecting the top FI-up port with the top FI-down port, sending a packet from the uplink ports, and verifying that the packet is received on the downlink ports.

Note:

Only run an external loopback test from the top FI ports, do not test FI cables with the bottom FI ports.

Always run the internal test first, because the FI cable tests are not conclusive until you ensure the FI ports work correctly.

Stack Health Check

The Stack Health Check feature provides information on the state of each units Fabric Interconnect port. Stack Health Check runs a high-level test to monitor the Fabric Interconnect port status for each unit, the number of units in a stack, if a stack runs with a temporary base-unit, and to monitor stack continuity.

Stack Health Monitoring and Recovery

The Stack Health Monitoring and Recovery feature provides failure detection and recovery of units in a stack. Stack Health Monitoring and Recovery consists of the following:

- Communication failure detection and recovery (Hello Module)
- Database exchange failure detection and recovery (Stack Manager)
- Synchronization failure detection and recovery (Stack Manager)
- VxWorks task suspension detection
- IMC buffer pool detection

Communication failure detection and recovery

The Hello Module provides communication failure detection and recovery. The Hello Module detects the presence of units connected to an FI-up or FI-down port, and periodically exchanges messages to determine if the neighboring units are functioning.

When a new unit is connected to an FI-up or FI-down port, the FI ports go into a link up state. The front panel UP or DOWN status LED state changes from off to amber in link up state. Amber indicates that the FI connection is physically up. When the Hello Module successfully exchanges a message sequence with a neighbor unit, the front panel UP or DOWN status LED state changes from amber to emerald. Emerald indicates that the FI connection is logically up.

If the Hello Module periodic message exchange with a neighbor unit fails, the FI-up or FI-down connection changes from logically up to logically down (physically up). If both FI-up and FI-

down are logically down for more than five minutes, a message is logged and the unit reboots. The Hello Module does not reboot a unit if an FI-port direction is logically up. You can enable or disable the Hello Module reboot on failure, the default is enabled.

Database exchange failure detection and recovery

The Stack Manager manages the exchange of database data between like applications running on multiple units in a stack. The Stack Manager on the base unit controls when an application can start a database data exchange. The base unit Stack Manager receives acknowledgement from each unit in the stack once an application successfully exchanges data. If an acknowledgement is not received, the Stack Manager attempts the database data exchange again, to a maximum of three attempts. If three successive database exchange attempts fail, all units in the stack become standalone units.

The Stack Manager can recover the stack after three successive database exchange failures. The Stack Manager waits three minutes after the database exchange failure, logs the failure, then restarts the stack formation process. You can configure the stack retry count. The default is 25.

Synchronization failure detection and recovery

After all applications successfully exchange database data, the Stack Manager on the base unit sends a stack synchronization message to all non-base units. The base unit Stack Manager receives acknowledgement from each unit in the stack. If an acknowledgement is not received, the Stack Manager sends the stack synchronization message again, to a maximum of three attempts. If three successive stack synchronization messages are not acknowledged, all units in the stack become standalone units.

The Stack Manager can recover the stack after three successive stack synchronization failures. The Stack Manager waits three minutes after the stack synchronization failure, logs the failure, then restarts the stack formation process. You can configure the stack retry count. The default is 25.

VxWorks task suspension detection

The Stack Monitor can detect a suspended task. The task status is checked every five minutes. If a task suspension is detected, the task name or indices is logged in the customer log.

IMC buffer pool leak detection

The Stack Monitor can detect a depleted IMC buffer pool. The IMC buffer pool is checked every five minutes. An IMC buffer pool is depleted if all buffers are allocated for two successive checks. If an IMC buffer pool is depleted, it is logged in the customer log.

Stack Monitor

The Stack Monitor can detect problems with the units in the stack and send traps. You can enable, disable, set the expected stack size, and control the frequency of trap sending.

The Stack Monitor sends a trap for the following events:

- The number of units in a stack changes.
- The trap sending timer expires.

Each time the number of units in a stack changes, the trap sending timer resets and the Stack Monitor compares the current number of stack units with the configured number of stack units. If the values are not equal, the switch sends a trap and logs a message to syslog. The Stack Monitor sends traps from a stand-alone unit or the base unit of the stack.

After the trap sending timer reaches the configured number of seconds at which traps are sent, the switch sends a trap and logs a message to syslog and restarts the trap sending timer. The syslog message is not repeated unless the stack configuration changes, this prevents the log from being filled with stack configuration messages.

Once you enable the Stack Monitor on a stack, the Stack Monitor captures the current stack size and uses it as the expected stack size. You can also configure a different stack size value when Stack Monitor is enabled.

Chapter 4: System diagnostics and statistics using ACLI

This section describes how to configure the ACLI commands used to diagnose your network and show statistics.

Viewing port statistics using ACLI

Use this procedure to view the statistics for the port on both received and transmitted traffic.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show port-statistics [port <portlist>]`

Variable definitions

The following table describes the parameters for the **`show port-statistics`** command.

Variable	Value
`port <portlist>`	Specifies the ports to display statistics for. If you do not specify a port, all ports are shown.

Job aid: viewing port statistics

The following table provides descriptions of statistical information that is displayed when you use the **`show port-statistics`** command for one or more switch ports.

Statistic	Description
Packets:	Indicates the total number of packets that were transmitted and received (including bad packets, broadcast packets, and multicast packets).
Multicasts:	Indicates the total number of good packets, transmitted and received, that were directed to a multicast address. This number does not include packets directed to the broadcast address.
Broadcasts:	Indicates the total number of good packets, transmitted and received, that were directed to the broadcast address. This does not include multicast packets.
Total Octets:	Indicates the total number of octets of data (including those in bad packets) that are transmitted and received on the network (excluding framing bits but including FCS octets). You can use this object as a reasonable estimate of Ethernet utilization.
Packets 64 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 1 and 64 octets in length (excluding framing bits but including FCS octets).
65–127 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 65 and 127 octets in length (excluding framing bits but including FCS octets).
128–255 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 128 and 255 octets in length (excluding framing bits but including FCS octets).
256–511 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 256 and 511 octets in length (excluding framing bits but including FCS octets).
512–1023 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 511 and 1023 octets in length (excluding framing bits but including FCS octets).
1024–1518 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 1024 and 1518 octets in length (excluding framing bits but including FCS octets).
1519–2047 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 1519 and 2047 octets in length (excluding framing bits but including FCS octets).

Statistic	Description
2048–4095 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 2048 and 4095 octets in length (excluding framing bits but including FCS octets).
4096–9216 bytes:	Indicates the total number of transmitted and received packets (including bad packets) that were between 4096 and 9216 octets in length (excluding framing bits but including FCS octets).
Jumbo:	Indicates the number of Ethernet frames containing more than 1518 bytes of data that were received on a port or ports.
Control Packets:	Indicates the number of control packets received on a port or ports.
FCS Errors:	Indicates the count of frames received on a particular interface that are an integral number of octets in length but do not pass the FCS check. The count represented by an instance of this object is incremented after the *frameCheckError* status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions occur are, according to the conventions of IEEE 802.3 Layer management, counted exclusively according to the error status presented to the LLC.
Undersized Packets:	Indicates the total number of received packets that were less than 64 octets long (excluding framing bits but including FCS octets) and were otherwise well formed.
Oversized Packets:	Indicates the total number of transmitted and received packets that were longer than 16383 octets (excluding framing bits but including FCS octets) and were otherwise well formed.
Filtered Packets:	Indicates the number of filtered packets received on a port or ports.
Pause Frames:	Indicates the number of pause frames transmitted from or received on a port or ports, due to network congestion.
Dropped On No Resources:	Indicates the number of transmitted packets that were dropped due to a lack of resources on a port or ports.

Displaying port operational status using ACLI

Use this procedure to display the port operational status.

Important:

If you use a terminal with a width of greater than 80 characters, the output is displayed in a tabular format.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show interfaces [<portlist>] [verbose]`

Note:

If you issue the command with no parameters the port status is shown for all ports.

Example

```
7024XLS#show interfaces 1/1
                    Status                    Auto                        Flow
Unit/Port Trunk Admin   Oper Link LinkTrap Negotiation Speed    Duplex Control
--------- ----- ------- ---- ---- -------- ----------- -------- ------ -------
1/1             Enable  Up   Up   Enabled  Enabled     1000Mbps Full   Symm

7024XLS#show interfaces 1/1 verbose
Unit/Port:  1/1
    Trunk:
    Admin Status:  Enable
    Oper Status:  Up
    VLACP Oper Status:  Down
    STP Oper Status:  Forwarding
    Link: Up
    LinkTrap:  Enabled
    Link Autonegotiation:  Enabled
    Link Speed:  1000Mbps
    Link Duplex:  Full-Duplex
    Flow Control:  Symm
    BPDU-guard (BPDU Filtering): Disabled
    BPDU-guard (BPDU Filtering) Oper Status: N/A

------------------------
```


Comments? infodev@avaya.com

Validating port operational status using ACLI

Use this procedure to show the port operational status for the following protocols:

- VLACP — Virtual Link Aggregation Control Protocol
- STP — Spanning Tree Protocol

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. After the switch boots, enter the following command to validate that STP status is Listening:
   `show interfaces <portlist> verbose`
3. Wait a few seconds and enter `show interfaces <portlist> verbose` to validate that the STP status changes to Learning.
4. Wait 15 seconds (the default time interval for STP forward delay), and then enter `show interfaces <portlist> verbose` again to validate that the STP status is Forwarding.
5. Configure VLACP on port 1 on a VSP 700 Series unit and on port 2 on another VSP 7000 Series unit. Have a link between these two ports. Enter `show interfaces <portlist> verbose` to validate that the VLACP status is up for port on the unit where the command is typed.
6. Pull out the link from the other switch and enter `show interfaces <portlist> verbose` to validate that the VLACP status goes Down.

Displaying port information using ACLI

Use this procedure to display port configuration information.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show interfaces <portlist> config`

Job aid

The following is an example of the `show interfaces <portlist> config` command.

```
7024XLS#show interfaces 1/1 config
Unit/Port:  1/1
    Trunk:
    Admin Status:  Enable
    Oper Status:  Up
    VLACP Oper Status:  Down
    STP Oper Status:  Forwarding
    Link: Up
    LinkTrap:  Enabled
    Link Autonegotiation:  Enabled
    Link Speed:  1000Mbps
    Link Duplex:  Full-Duplex
    Flow Control:  Symm
    BPDU-guard (BPDU Filtering): Disabled
    BPDU-guard (BPDU Filtering) Oper Status: N/A
```

```
*****VLAN interfaces configuration*****
           Filter     Filter
          Untagged Unregistered
Unit/Port  Frames     Frames    PVID PRI    Tagging    Name
--------- -------- ------------ ---- --- ------------- --------------
1/1       No       Yes          1    0   UntagAll      Unit 1,Port 1
```

```
*****VLAN ID port member configuration*****
Unit/Port VLAN VLAN Name         VLAN VLAN Name         VLAN VLAN Name
--------- ---- ----------------  ---- ----------------  ---- ----------------
1/1       1    VLAN #1
--------- ---- ----------------  ---- ----------------  ---- ----------------
```

```
*****Spanning-tree port configurations*****
Unit Port Trunk   Participation   Priority  Path Cost    State
---- ---- -----  --------------  --------  ---------  ----------
1    1           Normal Learning  128       1          Forwarding
```

Viewing environmental information using ACLI

Use this procedure to view the status of the unit or stack environment.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show environmental`

Comments? infodev@avaya.com

Job aid

The following is an example of the **show environmental** command output.

```
7024XLS-PWR#show environmental
Unit# PSU1      PSU2      FAN1 FAN2 FAN3 FAN4 Temperature
----- --------- --------- ---- ---- ---- ---- -----------
1     Primary   Redundant OK        OK   OK   OK   OK   32C
```

Perform the following steps to ensure that the unit or stack works in proper conditions.

1. Display the unit or stack environmental information from the ACLI.
2. If the information indicates that each units hardware environment is in good condition you have completed this procedure.
3. If the temperature is high or the fans have a fail status, check the hardware.
4. Execute hardware maintenance.
5. Repeat steps 1 to 4, if necessary.

Enabling the ACLI audit log using ACLI

Use this procedure to record and log all ACLI commands issued on the switch.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   audit log save
   ```

Disabling the ACLI audit log using ACLI

Use this procedure to discontinue recording and logging all ACLI commands issued on the switch.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:

```
no audit log
```

Displaying the ACLI audit log using ACLI

Use this procedure to display the ACLI audit log configuration status or to display the contents of the audit log for specific user connections to the switch.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show audit log [asccfg | serial | telnet | config]
   ```

Variable definitions

The following table describes the parameters for the **`show audit log`** command.

Variable	Value
asccfg	Displays the audit log contents for ASCII configuration commands.
serial	Displays the audit log contents for serial connections to the switch.
telnet	Displays the audit log contents for Telnet or SSH connections to the switch.
config	Displays whether saving the audit log to NVRAM is enabled or disabled.

Displaying Stack Health Check using ACLI

The Stack Health Check feature provides information on the state of each units Fabric Interconnect port. Stack Health Check runs a high-level test to monitor the Fabric Interconnect

port status for each unit, the number of units in a stack, if a stack runs with a temporary base-unit, and to monitor stack continuity.

Before you begin

Stack Health Check requires a stack configuration.

About this task

Use the following procedure to display the Stack Health Check information.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show stack health
   ```

Example

The following is an example of the `show stack health` command output when the stack is formed but did not finish the initialization process.

```
#show stack health
Stack in progress
```

The following is an example of the `show stack health` command output when the stack is formed and initialized, with all FI ports ok:

```
#show stack health
-------------------------------------------------------------
--
Unit#        Switch Model        Cascade Up        Cascade Down
-------------------------------------------------------------
1 (Base)     7024XLS                OK            OK
2         7024XLS                 OK                         OK
3         7024XLS                 OK                         OK
-------------------------------------------------------------
Switch Units Found = 3
Stack Health Check = OK - RESILIENT
Stack Diagnosis = Stack in full resilient mode
```

The following is an example of the `show stack health` command output when the stack is formed and initialized, with damaged or missing FI cables:

```
#show stack health
-------------------------------------------------------------
--
Unit#        Switch Model        Cascade Up        Cascade Down
-------------------------------------------------------------
1 (Base)     7024XLS                OK                   OK
2         7024XLS                 OK                LINK DOWN OR MISSING
3         7024XLS                 OK                      OK
-------------------------------------------------------------
Switch Units Found = 3
Stack Health Check = WARNING - NON-RESILIENT
Stack Diagnosis = Stack in non-resilient mode
Recommend to add/replace the identified cable(s).
```

The following is an example of the `show stack health` command output when the stack is formed and initialized, with an FI port malfunction:

```
#show stack health
-------------------------------------------------------------
--
Unit#        Switch Model        Cascade Up        Cascade Down
-------------------------------------------------------------
1 (Base)     7024XLS                OK                   OK
2         7024XLS                OK                 UP WITH ERRORS
3         7024XLS                OK                   OK
-------------------------------------------------------------
Switch Units Found = 3
Stack Health Check = WARNING - NON-RESILIENT
Stack Diagnosis = Stack in non-resilient mode
Recommend to add/replace the identified cable(s)
```

Note:

UP WITH ERRORS is not considered an FI cable problem until several status changes between OK and LINK DOWN occur in a short period of time.

Configuring the Stack Monitor using ACLI

Use the following procedure to configure the Stack Monitor.

Before you begin

Stack Monitor requires a stack configuration.

About this task

The Stack Monitor can detect problems with the units in the stack and send traps. You can enable, disable, default, set the expected stack size, and control the frequency of trap sending.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. To enable and configure the Stack Monitor, enter the following command:
   ```
   stack-monitor [enable] [stack-size <2-8>] [trap-interval <30-300>]
   ```

   Note:

   If you do not specify the stack-size or trap-interval parameters, the default values are used.
3. To default all Stack Monitor values, enter the following command:
   ```
   default stack-monitor
   ```
4. To disable Stack Monitor, enter the following command:
   ```
   no stack-monitor
   ```


Comments? infodev@avaya.com

Next steps

To view the current Stack Monitor status, use the following command:

```
show stack-monitor
```

Variable definitions

The following table describes the parameters for the **`stack-monitor`** command.

Variable	Value
enable	Enables Stack Monitor
stack-size <*2–8*>	Specifies the size of the stack to monitor. Valid range is 2 to 8. Default value is 2.
trap-interval <*30–300*>	Specifies the time interval between traps in seconds. Valid range is 30 to 300 seconds. Default value is 60 seconds.

Configuring Stack Health Monitoring and Recovery using ACLI

Use the following commands to configure Stack Health Monitoring and Recovery for Hello Module communication failure reboots, and Stack Manager retry counts.

Before you begin

Stack Health Monitoring and Recovery requires a stack configuration.

About this task

The Stack Health Monitoring and Recovery feature provides failure detection and recovery for stack communication, application database data exchange, and stack synchronization , as well as task suspension and buffer pool leak detection. You can enable or disable the stack reboot on communication failure. The Stack Manager can recover the stack after three successive database exchange or synchronization failures. You can configure the Stack Manager stack retry counter limit.

Note:

Stack reboot-on-failure is enabled by default. The Stack Manager default stack retry-count limit is 25.

Procedure

1. Log on to the Global Configuration mode in ACLI.

2. To enable the stack reboot on failure, enter the following command:
   ```
   stack reboot-on-failure
   ```
3. To disable the stack reboot on failure, enter the following command:
   ```
   no stack reboot-on-failure
   ```
4. To configure the Stack Manager retry count, enter the following command:
   ```
   stack retry-count [0 - 4294967295]
   ```

Next steps

You can use the following commands to display the current Stack Health Monitoring and Recovery configuration:

```
show stack reboot-on-failure
show stack retry-count
```

Viewing Stack Port Counters using ACLI

Use the following procedure to view or clear the Stack Port Counters.

Before you begin

Stack Port Counters requires a stack configuration.

About this task

Important:

The stack counters measure the size of packets received on FI-up and FI-down ports. The size of these packets is greater than the size of the packets received on front panel ports since an extra header is added to each of them to allow them to flow through the stack. The size of this header is 12 bytes, therefore another range of stack counters is incremented when sending packets having length close to the stack counters upper intervals limit.

The number of received/transmitted packets can be greater than the number of packets transmitted on front panel ports since different stack management packets are transmitted/received.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. To view the Stack Port Counters, enter the following command:
   ```
   show stack port-statistics [unit <1-8>]
   ```

3. To clear the Stack Port Counters, enter the following command:
   ```
   clear stack port-statistics [unit <1-8>]
   ```

Example

The following shows an example output from the `show stack port-statistics` command.

```
UP                      DOWN
-------------------------  -------------------  -------------------
Received
    Packets:                374                  378
    Multicasts:             374                  378
    Broadcasts:             0                    0
    Total Octets:           116738               117114
    Packets 64 bytes:       0                    0
            65-127 bytes:   198                  202
            128-255 bytes:  36                   36
            256-511 bytes:  12                   12
            512-1023 bytes: 92                   92
            1024-1518 bytes: 36                  36
            1519-2047 bytes: 0                   0
            2048-4095 bytes: 0                   0
            4096-9216 bytes: 0                   0
                 Jumbo     : 0                   0
    Control Packets:        0                    0
    FCS Errors:             0                    0
    Undersized Packets:     0                    0
    Oversized Packets:      0                    0
    Filtered Packets:       0                    0
    Pause Frames:           0                    0

Transmitted
    Packets:                1434                 1338
    Multicasts:             434                  338
    Broadcasts:             0                    0
    Total Octets:           1639915              1595645
    Packets 64 bytes:       0                    0
            65-127 bytes:   285                  245
            128-255 bytes:  15                   39
            256-511 bytes:  6                    18
            512-1023 bytes: 92                   0
            1024-1518 bytes: 36                  36
            1519-2047 bytes: 1000                1000
            2048-4095 bytes: 0                   0
            4096-9216 bytes: 0                   0
    Oversized Packets:      1000                 1000
    Pause Frames:           0                    0
    Dropped On No Resources: 0                   0
```

Testing Stack Loopback using ACLI

Use the following procedure to complete a stack loopback test.

Before you begin

The switch must be in stand alone mode to support Stack Loopback tests.

You must connect the top FI-up port to the top FI-down port on a switch with FI cables to test external loopback.

Note:

Only use the top FI-ports during an external loopback test.

About this task

You can quickly test your FI ports and FI cables by using the stack loopback test. The stack loopback test can determine whether the source of the problem is a defective FI cable or a damaged FI port. Two types of loopback tests exist. The internal loopback test verifies that the FI ports are functional. The external loopback test checks if the FI cables are functional. Always run the internal test first, because the FI cable tests are not conclusive until you ensure the FI ports work correctly.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. To run an internal loopback test, enter the following command:
   ```
   stack loopback-test internal
   ```
3. To run an external loopback test, enter the following command:
   ```
   stack loopback-test external
   ```

Example

The following examples show various loopback test results:

The Stack Loopback tests display the status of the FI ports and FI cables.

```
7024XLS#stack loopback-test internal
Testing uplink port … ok
Testing downlink port … ok
Internal loopback test PASSED.
7024XLS#
```

```
7024XLS#stack loopback-test external
External loopback test PASSED.
7024XLS#
```

The internal loopback test can detect a defective FI port.

```
7024XLS#stack loopback-test internal
Testing uplink port … Failed
Testing downlink port … ok
Internal loopback test FAILED.
7024XLS#
```

If the FI ports are functional, the external loopback test can detect a defective FI cable:

```
7024XLS#stack loopback-test external
External loopback test FAILED. Your stack cable might be damaged.
7024XLS#
```

If you run the command on any unit in a stack, the following error occurs:

```
7024XLS#stack loopback-test internal
Stack loopback test affects the functioning of the stack.
You should run this in stand-alone mode
```

```
7024XLS#stack loopback-test external
Stack loopback test affects the functioning of the stack. You
should run this in stand-alone mode
```

Chapter 5: Network monitoring configuration using ACLI

This section describes how to configure the ACLI commands used to monitor your network. Use these commands to view the CPU and memory utilization, check the system log, and configure port mirroring.

Viewing CPU utilization using ACLI

Use this procedure to view the switch CPU utilization.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show cpu-utilization`

Viewing memory utilization using ACLI

Use this procedure to view the switch memory utilization.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show memory-utilization`

Configuring the system log using ACLI

Use this procedure to configure the system settings for the system event log.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   logging [enable | disable] [level <critical | serious |
   informational | none>] [nv-level <critical | serious | none>]
   [volatile <latch | overwrite>]
   ```

Variable definitions

The following table describes the parameters for the **logging** command.

Variable	Value
`enable | disable`	Enables or disables the event log. DEFAULT: enable
`level <critical | serious | informational | none>`	Specifies the level of logging stored in DRAM.
`nv-level <critical | serious | none>`	Specifies the level of logging stored in NVRAM.
`volatile <latch | overwrite>`	Specifies the options for logging in DRAM.

Displaying the system log using ACLI

Use this procedure to displays the configuration and the current contents of the system event log.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show logging [config] [critical] [serious] [informational]
   [sort-reverse] [unit <1–8>]
   ```

Variable definitions

The following table describes the parameters for the **show logging** command.

Comments? infodev@avaya.com

Variable	Value
`config`	Displays configuration of event logging.
`critical`	Displays critical log messages.
`serious`	Displays serious log messages.
`informational`	Displays informational log messages.
`sort-reverse`	Displays informational log messages in reverse chronological order (beginning with most recent).
`unit <1-8>`	Displays log messages for a specific switch in a stack.

Disabling the system log using ACLI

Use this procedure to disable the system event log.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   `no logging`

Setting the system log to default using ACLI

Use this procedure to default the system event log configuration.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   `default logging`

Clearing the system log using ACLI

Use this procedure to clear all log messages in DRAM.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   clear logging [non-volatile] [nv] [volatile]
   ```

Variable definitions

The following table describes the parameters for the **`clear logging`** command.

Variable	Value
`non-volatile`	Clears log messages from NVRAM.
`nv`	Clears log messages from NVRAM and DRAM.
`volatile`	Clears log messages from DRAM.

Configuring remote system logging using ACLI

Use this procedure to configure and manage the logging of system messages on a remote server.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   logging remote [address <A.B.C.D> | <WORD>] [secondary-
   address <A.B.C.D> | <WORD>] [enable] [level <critical |
   informational | serious | none>] [facility <daemon | local0 |
   local1 | local2 | local3 | local4 | local5 | local6 | local7>]
   ```
3. To display the configuration and the current contents of the system event log, enter the following command:
   ```
   show logging
   ```

Variable definitions

The following table describes the parameters for the **`logging remote`** command.

Note:
The IPv6 parameter is valid only for switches that support IPv6.

Variable	Value
`address <A.B.C.D> | <WORD>`	Specifies the IP address of the primary remote system log server. • *A.B.C.D*—the IPv4 address of the remote server. • *WORD*—the remote host IPv6 address. The value is a character string with a maximum of 45 characters. DEFAULT: 0.0.0.0
`enable`	Enables the system message logging on remote server. **Important:** You must configure either the primary or secondary remote server address before you enable remote logging.
`level <critical | informational | serious | none>`	Specifies the remote logging level: • `critical`—only messages classified as critical are sent to the remote system log server. • `serious`—only messages classified as serious are sent to the remote system log server. • `informational`—only messages classified as informational are sent to the remote system log server. • `none`—no remote log messages are sent to the remote system log server. DEFAULT: `none`

Variable	Value
`secondary-address <A.B.C.D> | <WORD>`	Specifies the IP address of the secondary remote system log server. • *A.B.C.D*—the IPv4 address of the remote server. • *WORD*—the remote host IPv6 address. The value is a character string with a maximum of 45 characters. DEFAULT: 0.0.0.0
`facility <daemon | local0 | local1 | local2 | local3 | local4 | local5 | local6 | local7>`	Specifies the remote logging facility.

Disabling remote system logging using ACLI

Use this procedure to disable the logging of system messages on a remote server.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   `no logging remote [address] [secondary-address] [enable] [level]`

Variable definitions

The following table describes the parameters for the **`no logging remote`** command.

Variable	Value
`address`	Clears the primary remote system log server IP address.
`enable`	Disables system message logging on the remote server.
`level`	Clears the remote server logging level.
`secondary-address`	Clears the secondary remote system log server IP address.

Restoring remote system logging to default using ACLI

Use this procedure to restore the logging of system messages on a remote server to factory defaults.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   default logging remote [address] [secondary-address] [enable] [level]
   ```

Variable definitions

The following table describes the parameters for the **`default logging remote`** command.

Variable	Value
`address`	Restores the primary remote system log server IP address to the factory default (0.0.0.0).
`level`	Restores the remote server logging level to the factory default (none).
`secondary-address`	Restores the secondary remote system log server IP address to the factory default (0.0.0.0).

Performing a traceroute using ACLI

Use this procedure to display the route taken by IP packets to a specified host. For a stack configuration, you can preform this procedure only on the base unit.

About this task

Traceroute sends a sequence of User Datagram Protocol (UDP) probe packets to a specified host. You can adjust the time-to-live (TTL) IP parameter to determine the intermediate routers traversed. Traceroute displays a list of routers and the packet latency values during the route.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   traceroute <Hostname|A.B.C.D|WORD> <-m> <-p> <-q> <-v> <-w> <1-1464>
   ```
3. Type CTRL+C to interrupt the command.

Variable definitions

The following table describes the parameters for the **traceroute** command.

Variable	Value
Hostname	Specifies the name of the remote host.
A.B.C.D	Specifies the IP address of the remote host.
WORD	Specifies the IPv6 address of the remote host.
—m	Specifies the maximum time-to-live (TTL). The value range is 1–255. The default value is 10. Example: `traceroute 10.3.2.134 -m 10`
—p	Specifies the base UDP port number. The value range is 0–65535. The default value is 10. Example: `traceroute 10.3.2.134 -p 87`
—q	Specifies the number of probes per TTL. The value range is 1–255. The default value is 3. Example: `traceroute 10.3.2.134 -q 3`
—v	Specifies verbose mode. Example: `traceroute 10.3.2.134 -v`
—w	Specifies the wait time per UDP probe. The value range is 1–255. The default value is 5 seconds. Example: `traceroute 10.3.2.134 -w 5`
1–1464	Specifies the UDP probe packet size. The probe packet size is 40 plus the specified data length in bytes. Example: `traceroute 10.3.2.134 60`

Displaying IPv6 interface statistics using ACLI

Use this procedure to display IPv6 interface statistics.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 interface statistics [vlan <1-4094>]
   ```

Displaying IPv6 interface ICMP statistics using ACLI

Use this procedure to display IPv6 interface ICMP statistics.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 interface icmpstatistics [vlan <1-4094>]
   ```

Displaying IPv6 UDP statistics using ACLI

Use this procedure to display IPv6 UDP statistics.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 udp
   ```

Displaying IPv6 UDP endpoints using ACLI

Use this procedure to display IPv6 UDP endpoints.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 udp endpoints
   ```

Displaying IPv6 TCP statistics using ACLI

Use this procedure to display IPv6 TCP statistics.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 tcp
   ```

Displaying IPv6 TCP connections using ACLI

Use this procedure to display IPv6 TCP connections.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 tcp connections
   ```

Displaying IPv6 TCP listeners using ACLI

You can display IPv6 TCP listeners using ACLI.

About this task

Use the following procedure to show IPv6 TCP listeners.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show ipv6 tcp listeners
   ```

Comments? infodev@avaya.com

Chapter 6: RMON configuration using ACLI

This section describes the ACLI commands used to configure and manage RMON.

Viewing RMON alarms using ACLI

Use the following procedure to view RMON alarms.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show rmon alarm`

Viewing RMON events using ACLI

Use the following procedure to display information regarding RMON events.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   `show rmon event`

Viewing RMON ethernet history using ACLI

Use this procedure to display information regarding RMON ethernet history.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.

2. At the command prompt, enter the following command:
   ```
   show rmon ethernet history [delta | interval-range <hh:mm:ss>
   | port <portnumber> | sample-range <1–2147483647| sample-set
   <1–65535>]
   ```

Variable definitions

The following table describes the parameters for the **`show rmon ethernet history`** command.

Variable	Value
delta	Specifies the deltas of a consecutive history date.
interval-range<*hh:mm:ss*>	Specifies the history data for a specific interval range.
port<*portnumber*>	Specifies the history data for specific ports.
sample-range<*1–2147483674*>	Specifies the history data for a specific sample range.
sample-set<*1–65535*>	Specifies the history data for a specific index.

Viewing RMON ethernet statistics using ACLI

Use this procedure to display information regarding RMON ethernet statistics.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show rmon ethernet statistics [port <portnumber>]
   ```

Variable definitions

The following table describes the parameters for the **`show rmon ethernet statistics`** command.

Comments? infodev@avaya.com

Variable	Value
port<*portnumber*>	Displays the statistics for specific ports.

Viewing RMON history using ACLI

Use this procedure to display information regarding the configuration of RMON history.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show rmon history [<port>]
   ```

Variable definitions

The following table describes the parameters for the **`show rmon history`** command.

Variable	Value
port	Specifies the port number for the RMON history settings you want to display.

Viewing RMON statistics using ACLI

Use the following procedure to display information regarding the configuration of RMON statistics.

Procedure

1. Log on to the Privileged EXEC mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   show rmon stats
   ```

Setting RMON alarms using ACLI

Use the following procedure to set RMON alarms.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   rmon alarm <1–65535> <WORD> <1-2147483647> {absolute | delta}
   rising-threshold <-2147483648-2147483647> [<1–65535>]
   falling-threshold <-2147483648-2147483647> [<1–65535>]
   [owner <LINE>]
   ```

Variable definitions

The following table describes the parameters for the **`rmon alarm`** command.

Variable	Value
`<1-65535>`	Specifies a unique index for the alarm entry.
<*WORD*>	Specifies the MIB object to be monitored. This object identifier can be an English name.
`<1-2147483647>`	Specifies the sampling interval, in seconds.
`absolute`	Specifies the absolute values (value of the MIB object is compared directly with thresholds).
`delta`	Specifies the delta values (change in the value of the MIB object between samples is compared with thresholds).
`rising-threshold <-2147483648-21474836 47 > [<1-65535>]`	The first integer value is the rising threshold value. The optional second integer specifies the event entry to be triggered after the rising threshold is crossed. If omitted, or if an invalid event entry is referenced, no event is triggered.
`falling-threshold <-2147483648-21474836 47 > [<1-65535>]`	The first integer value is the falling threshold value. The optional second integer specifies the event entry to be triggered after the falling

Variable	Value
	threshold is crossed. If omitted, or if an invalid event entry is referenced, no event is triggered.
`[owner <LINE>]`	Specifies the owner string to identify the alarm entry.

Deleting RMON alarm table entries using ACLI

Use the following procedure to delete RMON alarm table entries.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   `no rmon alarm [<1–65535>]`

Variable definitions

The following table describes the parameters for the **`no rmon alarm`** command.

Variable	Value
`<1-65535>`	Specifies a unique index for the alarm entry. If you do not select a number, all RMON alarm table entries are deleted.

Configuring RMON event log and traps using ACLI

Use the following procedure to configure RMON event log and trap settings.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:

```
rmon event <1–65535> [log] [trap] [description <LINE>] [owner
<LINE>]
```

Variable definitions

The following table describes the parameters for the **rmon event** command.

Variable	Value
<1-65535>	Specifies a unique index for the alarm entry.
log	Records events in the log table.
trap	Generates SNMP trap messages for events.
description <LINE>	Specifies a textual description for the event.
owner <LINE>	Specifies the owner string to identify the alarm entry.

Deleting RMON event table entries using ACLI

Use the following procedure to clear entries in the table.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   no rmon event <1–65535>
   ```

Variable definitions

The following table describes the parameters for the **no rmon event** command.

Variable	Value
<1-65535>	Specifies a unique index for the event entry. If you do not specify an event entry, all table entries are deleted.

Configuring RMON history using ACLI

Use the following procedure to configure RMON history settings.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   `rmon history <1–65535> <LINE> <1–65535> <1–3600> [owner <LINE>]`

Variable definitions

The following table describes the parameters for the **`rmon history`** command.

Variable	Value
`<1-65535>`	Specifies a unique index for the history entry.
<LINE>	Specifies the port number to be monitored.
`trap`	Specifies the number of history buckets (records) to keep.
`description <LINE>`	Specifies the sampling rate (how often a history sample is collected).
`owner <LINE>`	Specifies the owner string to identify the history entry.

Deleting RMON history table entries using ACLI

Use this procedure to delete RMON history table entries.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:

```
no rmon history <1–65535>
```

Variable definitions

The following table describes the parameters for the **no rmon history** command.

Variable	Value
<1-65535>	Specifies a unique index for the history entry. If you do not specify an entry, all table entries are deleted.

Configuring RMON statistics using ACLI

Use this procedure to configure RMON statistics settings.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   rmon stats <1–65535> <LINE> [owner <LINE>]
   ```

Variable definitions

The following table describes the parameters for the **rmon stats** command.

Variable	Value
<1-65535>	Specifies a unique index for the stats entry.
<LINE>	Specifies the port number to configure.
owner <LINE>	Specifies the owner string to identify the stats entry.

Comments? infodev@avaya.com

Disabling RMON statistics using ACLI

Use this procedure to disable RMON statistics. If the variable is omitted, all entries in the table are cleared.

Procedure

1. Log on to the Global Configuration mode in ACLI.
2. At the command prompt, enter the following command:
   ```
   no rmon stats <1–65535>
   ```

Variable definitions

The following table describes the parameters for the **`no rmon stats`** command.

Variable	Value
`<1-65535>`	Specifies a unique index for the stats entry. If you do not specify an entry, all statistics are disabled.

Chapter 7: System diagnostics and statistics using Enterprise Device Manager

This chapter describes the procedures you can use to perform system diagnostics and gather statistics using Enterprise Device Manager (EDM).

Displaying switch summary in EDM

The EDM initial view displays a switch summary view in the work area. The Switch Summary tab displays basic switch information. This is an information-only display.

Procedure

1. From the navigation tree, double-click **Device**.
2. In the Device tree, click **Rediscover Device**.
3. On the toolbar, click the **Switch Summary** tab.

Variable definitions

The following table describes the variables associated with the switch summary tab.

Variable	Value
sysDescr	A description of the device.
sysUpTime	The time since the system booted.
sysObjectID	The system object identification number.
sysContact	Type the contact information (in this case, an e-mail address) for the system administrator.
sysName	Type the name of this device.
sysLocation	Type the physical location of this device.

Displaying switch temperature using EDM

Use the following procedure to display switch temperature information.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Environment**
4. In the work area, click **Temperature** tab.

Variable definitions

The following table describes the fields associated with Fan operating status.

Field	Description
Unit	Indicates the switch unit number in a stack. For a standalone switch, the default value is 1.
Temperature	Indicates the switch unit operating temperature.

Displaying switch fan information using EDM

Use this procedure to display information about the operating status of the switch fans

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Environment**
4. In the work area, click **Fan** tab.

Variable definitions

The following table describes the Fan operating status.

Field	Description
Unit 1 Fan 1	Indicates the status of Fan 1.
Unit 1 Fan 2	Indicates the status of Fan 2.
Unit 1 Fan 3	Indicates the status of Fan 3.
Unit 1 Fan 4	Indicates the status of Fan 4.

Displaying power supply information using EDM

Use this procedure to display the operating status of switch power supplies.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Environment**
4. In the Environment work area, click **PowerSupply** tab.

Variable definitions

Use the data in the following table to help you understand the switch power supply display.

Field	Description
Unit 1 Power Supply 1	Primary power supply.
Unit 1 Power Supply 2	Redundant power supply..
OperState	Indicates the status of the applicable power supply.

Displaying stack health using EDM

Use this procedure to display stack health information.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Switch/Stack**.
4. In the work area, click the **Stack Health** tab.

Variable definitions

The following table describes the variables associated with Stack Health

Variable	Value
Switch Units Found	Indicates the number of switch units in the stack.
Stack Health Check	Indicates the stack health.
Stack Diagnosis	Indicates the stack mode.

Displaying stack information using EDM

Use this procedure to display information about the switch units in the stack.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Switch/Stack**
4. In the work area, click the **Stack Info** tab.
5. Each row represents a different switch unit in the stack.

Variable definitions

This table provides descriptions of the variables on the Stack Info page.

Variable	Value
Indx	Indicates the line number for stack info. This is a read-only cell.
Descr	Describes the component or subcomponent. If not available, the value is a zero length string. This is a read-only cell.
Location	Specifies the geographic location of a component in a system modeled as a chassis, but possibly physically implemented with geographically separate devices connected to exchange management information. Chassis modeled in this manner are sometimes referred to as virtual chassis. An example value is: 4th flr wiring closet in blg A. **Important:** This field applies only to components that are in either the Board or Unit groups. If the information is unavailable, for example, the chassis is not modeling a virtual chassis or component is not in a Board or Unit group, the value is a zerolength string. If this field is applicable and is not assigned a value through a SNMP SET PDU when the row is created, the value defaults to the value of the object s5ChasComSerNum.
LstChng	Indicates the value of sysUpTime when the component or subcomponent is added to the stack. If this action did not occur since the cold or warm start of the agent, the value is zero. This is a read-only cell.
AdminState	Specifies the state of the component or subcomponent. • enable: enables operation • reset: resets component
OperState	Indicates the current operational state of the component. This is a read-only cell. Values include:

Variable	Value
	• other: another state • notAvail: state not available • removed: component removed • disabled: operation disabled • normal: normal operation • resetInProg: reset in progress • testing: performing a self test • warning: operating at warning level • nonFatalErr: operating at error level • fatalErr: error stopped operation The component type determines the allowable (and meaningful) values.
Ver	Indicates the version number of the component or subcomponent. If not available, the value is a zero-length string. This is a read-only cell.
SerNum	Indicates the serial number of the component or subcomponent. If not available, the value is a zero-length string. This is a read-only cell.
BaseNumPorts	Indicates the number of base ports of the component or subcomponent. This is a read-only cell.
TotalNumPorts	Indicates the number of ports of the component or subcomponent. This is a read-only cell.
IpAddress	Indicates the IP address of the component or subcomponent. This is a read-only cell.
RunningSoftwareVer	Indicates the software version running on the switch. This is a read-only cell.

Editing switch unit location in the stack using EDM

Use this procedure to change the location of the switch units in the stack.

The switch unit location refers to the geographic location of the unit in the system chassis. The field applies only to components that are in either the Board or Unit groups.

An example value is: 4th flr wiring closet in blg A.

Note:

If the information is unavailable, for example, the chassis is not modeling a virtual chassis or component is not in a Board or Unit group, the value is a zerolength string. If this field is applicable and is not assigned a value through a SNMP SET PDU when the row is created, the value defaults to the value of the object s5ChasComSerNum.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Switch/Stack**
4. In the work area, click **Stack Info** .
5. To select a switch unit for which to edit information, click a switch row.
6. In the row, double-click the cell in the **Location** column.
7. Type a location.
8. On the toolbar, click **Apply**.

Editing the switch AdminState in the stack using EDM

Use this procedure to enable or reset the switch AdminState in the stack. You can specify the state of the component or subcomponent.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Switch/Stack**
4. In the work area, click the **Stack Info** tab.
5. Each row represents a different switch unit in the stack.
6. In the row, double-click the cell in the **AdminState** column.
7. Select a value from the list (either enable or reset).
8. On the toolbar, click **Apply**.

Configuring Stack Monitor using EDM

Use the following procedure to configure the Stack Monitor.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Chassis**.
4. In the work area, click the **Stack Monitor** tab.
5. Select **StackErrorNotificationEnabled** to enable stack monitoring.
6. Set the stack size you want to monitor in the **ExpectedStackSize** field.
7. Set the traps interval in the **StackErrorNotificationInterval** field.
8. On the toolbar, click **Apply**.

Variable definitions

The following table describes the fields associated with the Stack Monitor tab.

Field	Description
StackErrorNotificationEnabled	Enables or disables the Stack Monitoring feature.
ExpectedStackSize	Sets the size of the stack to monitor. Valid range is 2–8.
StackErrorNotificationInterval	Sets the interval between traps, in seconds. Valid range is 30 to 300 seconds.

Creating a graph using EDM

Several screens in the EDM provide a means to view and make use of statistical information gathered by the switch.

Use the following procedure to turn this statistical information in either a bar, line, area, or pie graph.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis** or **Port**.
3. In either the Chassis or Port tree, click the tab that provides the data that you want to graph.
4. Select the desired graph type by clicking the applicable diagram.

Graphing chassis IP statistics using EDM

Perform this procedure to display and graph switch IP statistics.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis**.
3. In the Chassis work area, click **IP** tab.
4. On the toolbar, select a **Poll Interval** from the drop down menu.
5. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
6. On the toolbar, click **Line Chart, Area Chart, Bar Chart** or **Pie Chart**.

Variable definitions

Use the data in the following table to help you understand IP statistics.

Variables	Value
InReceives	The total number of input datagrams received from interfaces, including those received in error.
InHdrErrors	The number of input datagrams discarded due to errors in their IP headers, including bad checksums, version number mismatch, other format errors, time-to-live exceeded, errors discovered in processing their IP options.

Variables	Value
InAddrErrors	The number of input datagrams discarded because the IP address in the IP header destination field is not valid. This count includes invalid addresses (for example, 0.0.0.0) and addresses of unsupported Classes (for example, Class E). For addresses that are not IP Gateways and therefore do not forward datagrams, the counter includes datagrams discarded because the destination address is not a local address.
ForwDatagrams	The number of input datagrams for which this entity is not their final IP destination, and an attempt is made to find a route to forward them to the final destination. For addresses that do not act as IP Gateways, the counter includes only those packets Source-Routed by way of this address with successful Source-Route option processing.
InUnknownProtos	The number of locally addressed datagrams received successfully but discarded because of an unknown or unsupported protocol.
InDiscards	The number of input IP datagrams for which no problems are encountered to prevent their continued processing, but that are discarded (for example, for lack of buffer space). This counter does not include any datagrams discarded while awaiting reassembly.
InDelivers	The total number of input datagrams successfully delivered to IP user-protocols (including ICMP).
OutRequests	The total number of IP datagrams that local IP user-protocols (including ICMP) supplied to IP in requests for transmission. This counter does not include any datagrams counted in ipForwDatagrams.
OutDiscards	The number of output IP datagrams where no problem is encountered to prevent their transmission to the destination, but that are discarded (for example, for lack of buffer space). The counter can include datagrams counted in ipForwDatagrams if any such packets met this (discretionary) discard criterion.

Variables	Value
OutNoRoutes	The number of IP datagrams discarded because no route can be found to transmit them to their destination. This counter also includes any packets counted in ipForwDatagrams that have no route. This includes any datagrams a host cannot route because all of its default gateways are down.
FragOKs	The number of IP datagrams successfully fragmented at this entity.
FragFails	The number of IP datagrams discarded because they need to be fragmented at this entity but cannot be, for example, because the Don't Fragment flag is set.
FragCreates	The number of generated IP datagram fragments because of a fragmentation at this entity.
ReasmReqds	The number of IP fragments received that needed to be reassembled at this entity.
ReasmOKs	The number of IP datagrams successfully reassembled.
ReasmFails	The number of failures detected by the IP reassembly algorithm (for example, timed out, errors). This is not necessarily a count of discarded IP fragments because some algorithms (notably the algorithm in RFC815) can lose track of the number of fragments by combining them as they are received.

Graphing chassis ICMP In Statistics using EDM

Use this procedure to display and graph ICMP In statistics. Using the information provided in the summary statistics window, you can identify network traffic behavior and help isolate network configuration and performance problems.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis**.
3. In the Chassis work area, click the **ICMP In** tab.

4. On the toolbar, select a **Poll Interval** from the dropdown menu.
5. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
6. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable Definitions

Use the data in the following table to help you understand the variables associated with ICMP In statistics.

Variable	Value
SrcQuenchs	The number of ICMP Source Quench messages received.
Redirects	The number of ICMP Redirect messages received.
Echos	IThe number of ICMP Echo (request) messages received.
EchoReps	The number of ICMP Echo Reply messages received.
Timestamps	The number of ICMP Timestamp (request) messages received.
TimestampReps	The number of ICMP Timestamp Reply messages received.
AddrMasks	The number of ICMP Address Mask Request messages received.
AddrMaskReps	The number of ICMP Address Mask Reply messages received.
ParmProbs	The number of ICMP Parameter Problem messages received.
DestUnreachs	The number of ICMP Destination Unreachable messages received.
TimeExcds	The number of ICMP Time Exceeded messages received.

Graphing chassis ICMP Out statistics using EDM

Use this procedure to display and graph ICMP Out statistics. Using the information provided in the summary statistics window, you can identify network traffic behavior and help isolate network configuration and performance problems.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis**.
3. In the Chassis work area, click the **ICMP Out** tab.
4. On the toolbar, select a**Poll Interval** from the dropdown menu.
5. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
6. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart.**

Variable Definitions

Use the data in the following table to help you understand ICMP Out statistics.

Variable	Value
SrcQuenchs	The number of ICMP Source Quench messages sent.
Redirects	The number of ICMP Redirect messages received. For a host, this object is always zero because hosts do not send redirects.
Echos	The number of ICMP Echo (request) messages sent.
EchoReps	The number of ICMP Echo Reply messages sent.
Timestamps	The number of ICMP Timestamp (request) messages sent.
TimestampReps	The number of ICMP Timestamp Reply messages sent.
AddrMasks	The number of ICMP Address Mask Request messages sent.

Variable	Value
AddrMaskReps	The number of ICMP Address Mask Reply messages sent.
ParmProbs	The number of ICMP Parameter Problem messages sent.
DestUnreachs	The number of ICMP Destination Unreachable messages sent.
TimeExcds	The number of ICMP Time Exceeded messages sent.

Graphing chassis TCP statistics using EDM

Use this procedure to display and graph TCP statistics. Graphing TCP statistics provides information about TCP segments, including details about timeouts, the number of maximum connections, active and passive connections, failed and reset connections, received segments, sent and retransmitted segments and received errors.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis**.
3. In the Chassis work area, click **TCP** tab.
4. On the toolbar, select a **Poll Interval** from the dropdown menu.
5. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
6. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

Use the data in the following table to help you understand TCP statistics.

Variable	Value
ActiveOpens	The number of times TCP connections make a direct transition to the SYN-SENT state from the CLOSED state.
PassiveOpens	The number of times TCP connections make a direct transition to the SYN-RCVD state from the LISTEN state.

Variable	Value
AttemptFails	The number of times TCP connections make a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections make a direct transition to the LISTEN state from the SYNRCVD state.
EstabResets	The number of times TCP connections make a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.
CurrEstab	The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.
InSegs	The total number of segments received, including those received in error. This count includes segments received on currently established connections.
OutSegs	The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.
RetransSegs	The total number of segments retransmitted, that is, the number of TCP segments transmitted containing one or more previously transmitted octets.
InErrs	The total number of segments received in error (for example, bad TCP checksums).
OutRsts	The number of TCP segments sent containing the RST flag.
HCInSegs	The number of segments received, including those received in error. This count includes segments received on currently established connections. This object is the 64-bit equivalent of InSegs.
HCOutSegs	The number of segments sent, including those on current connections, but excluding those containing only retransmitted octets. This object is the 64-bit equivalent of OutSegs.

Graphing chassis UDP statistics using EDM

Use this procedure to display and graph UDP statistics.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis**.
3. In the Chassis work area, click **UDP** tab.
4. On the toolbar, select a **Poll Interval** from the dropdown menu.
5. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
6. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

Use the data in the following table to understand the UDP statistic variables.

Variable	Value
InDatagrams	The total number of UDP datagrams delivered to UDP users.
NoPorts	The total number of received UDP datagrams for which there is no application at the destination port.
InErrors	The number of received UDP datagrams that cannot be delivered for reasons other than the lack of an application at the destination port.
OutDatagrams	The total number of UDP datagrams sent from this entity.
HCInDatagrams	The number of TCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.
HCOutDatagrams	The number of UDP datagrams sent from this entity, for devices that can transmit more than 1 million UDP datagrams for each second. Discontinuities in the value of this counter can occur at reinitialization of the

Comments? infodev@avaya.com

Variable	Value
	management system, and at other times as indicated by discontinuities in the value of sysUpTime.

Graphing chassis LACP diagnostic statistics using EDM

Use the following procedure to graph Link Aggregation Control Protocol (LACP) diagnostic statistics.

Procedure

1. From the **Device Physical View**, select a port.
2. From the navigation tree, click **Graph**.
3. In the Graph tree, click **Port**.
4. In the work area, click the **LACP** tab.
5. On the toolbar, select a **Poll Interval** from the dropdown menu.
6. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
7. On the toolbar, select **Line Chart, Area Chart, Bar Chart,** or **Pie Chart** by clicking the applicable icon.

Variable definitions

The following table describes the fields of LACP tab.

Field	Description
LACPDUsRX	Denotes the number of valid LACPDUs received on this Aggregation Port. This value is read-only.
MarkerPDUsRX	Signifies the number of valid Marker PDUs received on this Aggregation Port. This value is read-only.
MarkerResponsePDUsRX	The number of valid Marker Response PDUs received on this Aggregation Port. This value is read-only.
UnknownRX	Indicates the number of frames received that can

Field	Description
	• Carry the Slow Protocols Ethernet Type value (43B.4), but contain an unknown PDU. • Are addressed to the Slow Protocols group MAC Address (43B.3), but do not carry the Slow Protocols Ethernet Type. This value is read-only.
IllegalRX	Denotes the number of frames received that carry the Slow Protocols Ethernet Type value (43B.4), but contain a badly formed PDU or an illegal value of Protocol Subtype (43B.4). This value is read-only.
LACPDUsTX	Signifies the number of LACPDUs that are transmitted on this Aggregation Port. This value is read-only.
MarkerPDUsTX	Displays the number of Marker PDUs transmitted on this Aggregation Port. This value is read-only.
MarkerResponsePDUsTX	Indicates the number of Marker Response PDUs that are transmitted on this Aggregation Port. This value is read-only.

Graphing chassis SNMP statistics using EDM

Use the following procedure to statistical information about SNMP traffic.

The SNMP tab provides information about outgoing and incoming traffic on the switch.

Procedure

1. From the navigation tree, click **Graph**.
2. In the Graph tree, click **Chassis**.
3. In the work area, click the **SNMP** tab.

Comments? infodev@avaya.com

Variable definitions

The following table describes the fields of SNMP tab.

Field	Description
InPkts	Indicates the total number of messages delivered to the SNMP from the transport service.
OutPkts	Indicates the total number of SNMP messages passed from the SNMP protocol to the transport service.
InTotalReqVars	Indicates the total number of MIB objects retrieved successfully by the SNMP protocol as the result of receiving valid SNMP Get-Request and Get-Next PDUs.
InTotalSetVars	Indicates the total number of MIB objects altered successfully by the SNMP protocol as the result of receiving valid SNMP Set-Request PDUs.
InGetRequests	Indicates the total number of SNMP Get-Request PDUs that have been accepted and processed by the SNMP protocol.
InGetNexts	Indicates the total number of SNMP Get-Next PDUs accepted and processed by the SNMP protocol.
InSetRequests	Indicates the total number of SNMP Set-Request PDUs accepted and processed by the SNMP protocol.
InGetRequests	Indicates the total number of SNMP Get-Response PDUs accepted and processed by the SNMP protocol.
OutTraps	Indicates the total number of SNMP Trap PDUs generated by the SNMP protocol.
OutTooBigs	Indicates the total number of SNMP PDUs generated by the SNMP protocol for which the value of the error-status field is tooBig.
OutNoSuchNames	Indicates the total number of SNMP PDUs generated by the SNMP protocol for which the value of the error-status field is noSuchName.

Field	Description
OutBadValues	Indicates the total number of SNMP PDUs generated by the SNMP protocol for which the value of the error-status field is badValue.
OutGenErrs	Indicates the total number of SNMP PDUs generated by the SNMP protocol for which the value of the error-status field is genErr.
InBadVersions	Indicates the total number of SNMP messages delivered to the SNMP protocol for an unsupported SNMP version.
InBadCommunityNames	Indicates the total number of SNMP messages delivered to the SNMP protocol that used an unknown SNMP community name.
InBadCommunityUses	Indicates the total number of SNMP messages delivered to the SNMP protocol that represented an SNMP operation not allowed by the SNMP community named in the message.
InASNParseErrs	Indicates the total number of ASN.1 or BER errors encountered by the SNMP protocol after decoding received SNMP messages.
InTooBigs	Indicates the total number of SNMP PDUs delivered to the SNMP protocol for which the value of the error-status field is tooBig.
InNoSuchNames	Indicates the total number of SNMP PDUs delivered to the SNMP protocol for which the value of the error-status field is noSuchName.
InBadValues	Indicates the total number of SNMP PDUs delivered to the SNMP protocol for which the value of the error-status field is badValue.
InReadOnlys	Indicates the total number of SNMP PDUs delivered to the SNMP protocol for which the value of the error-status field is readOnly. This error is a protocol error to generate an SNMP PDU containing the value "readOnly" in the errorstatus field. This object is provided to detect incorrect implementations of the SNMP.
InGenErrs	Indicates the total number of SNMP PDUs delivered to the SNMP protocol for which the value of the error-status field is genErr.

Graphing chassis bridge statistics using EDM

Use the following procedure to graph port frame statistics on the chassis bridge.

Procedure

1. From the **Device Physical View**, select one or multiple ports.
2. From the navigation tree, click **Graph**.
3. In the Graph tree, click **Port**.
4. In the work area, click the **Bridge** tab.
5. On the toolbar, select a **Poll Interval** from the dropdown menu.
6. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
7. On the toolbar, select **Line Chart, Area Chart, Bar Chart,** or **Pie Chart** by clicking the applicable icon.

Variable definitions

The following table describes the fields of Bridge tab.

Field	Description
DelayExceededDiscards	Indicates the number of frames discarded by the port due to excessive transit delays through the bridge, incremented by both transparent and source route bridges.
MtuExceededDiscards	Indicates the number of frames discarded by the port due to an excessive size, incremented by both transparent and source route bridges.
InFrames	Indicates the number of frames that have been received by this port from its segment.
OutFrames	Indicates the number of frames that have been received by this port from its segment.
InDiscards	Indicates the count of valid frames received which were discarded (filtered) by the Forwarding Process.

Displaying unit information using EDM

Use this procedure to display the Unit information for a switch.

Procedure

1. In the **Device Physical View**, select the unit.
2. From the navigation tree, double-click **Edit**.
3. In the Edit tree, click **Unit**.
4. In the work area, click the **Unit** tab.

Variable definitions

The following table describes the variables associated with the unit tab.

Variable	Value
Type	Specifies the type number.
Descr	Specifies the type number.
Ver	Specifies the version number of the switch.
SerNum	Specifies the serial number of the switch.
BaseNumPorts	Specifies the base number of ports.
TotalNumPorts	Specifies the total number of ports.

Managing unit statistics using EDM

Use the following procedures to display and graph unit statistics using EDM.

Displaying unit statistics using EDM

Use the following procedure to view the statistical information of a unit.

Procedure

1. In the **Device Physical View**, select the unit.
2. From the navigation tree, double-click **Edit**.
3. In the Edit tree, click **Unit**.
4. In the Unit work area, click the **Unit Stats** tab.

Variable definitions

The following table describes the variables associated with unit statistics.

Variable	Value
Absolute Value	Indicates the counter value of packets dropped for the unit.
Cumulative	Indicates the total value of packets dropped seen since dialog displayed.
Average/sec	Indicates the average value of packets dropped per second.
Minimum/sec	Indicates the smallest value of packets dropped seen per second.
Maximum/sec	Indicates the largest value of packets dropped seen per second.
LastVal/sec	Indicates the last value of packets dropped seen per second.

Graphing unit statistics using EDM

Use the following procedure to graph the statistics associated with the unit.

Procedure

1. From the navigation tree, double-click **Edit**.
2. In the Edit tree, click **Unit**.
3. In the Unit work area, click the **Unit Stats** tab.
4. On the toolbar, click **Clear Counters**.
5. Select a **Poll Interval** from the dropdown menu.
6. In the Unit Stats section, select a data column to graph.

7. On the toolbar, select **Line, Area, Bar or Pie chart**.

Graphing port interface statistics using EDM

Use this procedure to display and graph interface parameters for a port.

Procedure

1. On the Device Physical view, click a port
2. From the navigation tree, click **Graph**.
3. In the Graph tree, click **Port**.
4. In the Port work area, click the **Interface** tab.
5. On the toolbar, select a **Poll Interval** from the dropdown menu.
6. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
7. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

Use the data in the following table to help you understand interface statistics.

Variable	Value
InOctets	The total number of octets received on the interface, including framing characters.
OutOctets	The total number of octets transmitted out of the interface, including framing characters.
InUcastPkts	The number of packets delivered by this sublayer to a higher sublayer that are not addressed to a multicast or broadcast address at this sublayer.
OutUcastPkts	The total number of packets that higher-level protocols requested be transmitted, and that are addressed to a multicast or broadcast address at this sublayer, including those that are discarded or not sent.
InMulticastPkts	The number of packets delivered by this sublayer to a higher sublayer that were

Comments? infodev@avaya.com

Variable	Value
	addressed to a multicast address at this sublayer. For a MAC layer protocol, this number includes both group and functional addresses.
OutMulticastPkts	The number of packets that higher-level protocols requested be transmitted, and that are addressed to a multicast address at this sublayer, including those that were discarded or not sent. For a MAC layer protocol, this number includes both group and functional addresses.
InBroadcastPkts	The number of packets delivered by this sublayer to a higher sublayer that are addressed to a broadcast address at this sublayer.
OutBroadcastPkts	The number of packets that higher-level protocols requested be transmitted, and that were addressed to a broadcast address at this sublayer, including those that were discarded or not sent.
InDiscards	The number of inbound packets chosen to be discarded even though no errors were detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet can be to free up buffer space.
OutDiscards	The number of outbound packets chosen to be discarded even though no errors were detected to prevent their being transmitted. One possible reason for discarding such a packet can be to free up buffer space.
InErrors	For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol.
OutErrors	For packet-oriented interfaces, the number of outbound packets that cannot be transmitted because of errors. For character oriented or fixed-length interfaces, the number of outbound transmission units that cannot be transmitted because of errors.

Variable	Value
InUnknownProtos	For packet-oriented interfaces, the number of packets received through the interface that are discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing, the number of transmission units received through the interface that are discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter is always zero.

Graphing port Ethernet error statistics using EDM

Use this procedure to display and graph Ethernet error statistics.

Procedure

1. On the Device Physical View, click a port.
2. From the navigation tree, click **Graph**.
3. In the Graph tree, click **Port**.
4. In the Port work area, click **Ethernet Errors** tab.
5. On the toolbar, select a **Poll Interval** from the dropdown menu.
6. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
7. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

Use the data in the following table to help you understand the Ethernet error statistics.

Variable	Value
AlignmentErrors	A count of frames received on a particular interface that are not an integral number of octets in length and do not pass the FCS check. The count represented by an instance of this object is incremented when the AlignmentError status is returned by the MAC service to the LLC (or other MAC user).

Variable	Value
	Received frames for which multiple error conditions occur are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.
FCSErrors	A count of frames received on a particular interface that are an integral number of octets in length, but do not pass the FCS check. The count represented by an instance of this object is incremented when the FCSErrors status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions occur are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.
InternalMacTransmitErrors	A count of frames for which transmission on a particular interface fails due to an internal MAC sublayer transmit error. A frame is only counted by an instance of this object if it is not counted by the corresponding instance of either the LateCollisions object, the ExcessiveCollisions object, or the CarrierSenseErrors object.
InternalMacReceiveErrors	A count of frames for which reception on a particular interface fails due to an internal MAC sublayer receive error. A frame is only counted by an instance of this object if it is not counted by the corresponding instance of either the FrameTooLongs object, the AlignmentErrors object, or the FCSErrors object. The precise meaning of the count represented by an instance of this object is implementation specific. In particular, an instance of this object can represent a count of receive errors on a particular interface that are not otherwise counted.
CarrierSenseErrors	The number of times that the carrier sense condition is lost or never asserted when attempting to transmit a frame on a particular interface. The count represented by an instance of this object increments once for each transmission attempt, even if the carrier sense condition fluctuates during a transmission attempt.

Variable	Value
FrameTooLongs	A count of frames received on a particular interface that exceed the maximum permitted frame size. The count represented by an instance of this object is incremented when the FrameTooLongs status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions occur are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.
SQETestErrors	A count of times that the SQE Test Errors message is generated by the PLS sublayer for a particular interface. The SQE TEST ERROR message is defined in section 7.2.2.2.4 of ANSI/IEEE 802.3-1985 and its generation is described in section 7.2.4.6 of the same document.
DeferredTransmissions	A count of frames for which the first transmission attempt on a particular interface is delayed because the medium is busy. The count represented by an instance of this object does not include frames involved in collisions.
SingleCollisionFrames	A count of successfully transmitted frames on a particular interface for which transmission is inhibited by exactly one collision. A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts, ifOutMulticastPkts, or ifOutBroadcastPkts,and is not counted by the corresponding instance of the MultipleCollisionFrames object.
MultipleCollisionFrames	A count of successfully transmitted frames on a particular interface for which transmission is inhibited by more than one collision. A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts, ifOutMulticastPkts, or ifOutBroadcastPkts, and is not counted by the corresponding instance of the SingleCollisionFrames object.
LateCollisions	The number of times that a collision is detected on a particular interface later than

Variable	Value
	512 bit-times into the transmission of a packet. Five hundred and twelve bit-times corresponds to 51.2 microseconds on a 10 Mb/s system. A (late) collision included in a count represented by an instance of this object is also considered as a (generic) collision for purposes of other collision-related statistics.
ExcessiveCollisions	A count of frames for which transmission on a particular interface fails due to excessive collisions.

Graphing port RMON statistics using EDM

Use this procedure to display and graph RMON Ethernet statistics.

Procedure

1. On the Device Physical View, click a port.
2. From the navigation tree, click **Graph**.
3. In the Graph tree, click **Port**.
4. Click the **RMON** tab.
5. On the toolbar, select a **Poll Interval** from the dropdown menu.
6. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
7. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

Use the data in the following table understand RMON Ethernet statistics.

Variable	Value
Octets	The total number of octets of data (including those in bad packets) received on the network (excluding framing bits but including FCS octets). You can use this object as a reasonable estimate of Ethernet utilization. For greater precision, sample the

Variable	Value
	etherStatsPkts and etherStatsOctets objects before and after a common interval.
Pkts	The total number of packets (including bad packets, broadcast packets, and multicast packets) received.
BroadcastPkts	The total number of good packets received that are directed to the broadcast address. This does not include multicast packets.
MulticastPkts	The total number of good packets received that are directed to a multicast address. This number does not include packets directed to the broadcast address.
CRCAlignErrors	The total number of packets received with a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, but with either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error).
UndersizePkts	The total number of packets received that are less than 64 octets long (excluding framing bits but including FCS octets) and were otherwise well formed.
OversizePkts (>1518)	The total number of packets received that are longer than 1518 octets (excluding framing bits but including FCS octets) and were otherwise well formed.
Fragments	The total number of packets received that are less than 64 octets in length (excluding framing bits but including FCS octets) and with either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error). For etherStatsFragments to increment is normal because it counts both runts (which are normal occurrences due to collisions) and noise hits.
Collisions	The best estimate of the total number of collisions on this Ethernet segment.
Jabbers	The total number of packets received that are longer than 1518 octets (excluding framing bits, but including FCS octets), with either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral

Variable	Value
	number of octets (Alignment Error). Jabber is defined as the condition where any packet exceeds 20 ms. The allowed range to detect jabber is between 20 ms and 150 ms.
1..64	The total number of packets (including bad packets) received that are less than or equal to 64 octets in length (excluding framing bits but including FCS octets).
65..127	The total number of packets (including bad packets) received that are greater than 64 octets in length (excluding framing bits but including FCS octets).
128..255	The total number of packets (including bad packets) received that are greater than 127 octets in length (excluding framing bits but including FCS octets).
256..511	The total number of packets (including bad packets) received that are greater than 255 octets in length (excluding framing bits but including FCS octets).
512..1023	The total number of packets (including bad packets) received that are greater than 511 octets in length (excluding framing bits but including FCS octets).
1024..1518	The total number of packets (including bad packets) received that are greater than 1023 octets in length (excluding framing bits but including FCS octets).

The following table outlines the statistics displayed for the RMON.

Statistic	Description
Poll Interval	Statistics are updated based on the poll interval. The default value is 10s. Valid range is None, 2s, 5s, 10s, 30s, 1m, 5m, 30m 1h.
Absolute	Indicates the total count since the last time counters were reset. A system reboot resets all counters.
Cumulative	Indicates the total count since the statistics tab opened. The elapsed time for the cumulative counter is shown at the bottom of the graph window.

Statistic	Description
Average/sec	Indicates the cumulative count divided by the cumulative elapsed time.
Minimum/sec	Indicates the minimum average for the counter for a polling interval over the cumulative elapsed time.
Maximum/sec	Indicates the maximum average for the counter for a polling interval over the cumulative elapsed time.
LastVal/sec	Indicates the average for the counter over the last polling interval.

Graphing miscellaneous port statistics using EDM

Use this procedure to display and graph miscellaneous statistics for a switch port.

Procedure

1. On the Device Physical View, click a port.
2. From the navigation tree, click **Graph**.
3. In the Graph tree, click **Port**.
4. In the Port work area, click the**Misc.** tab.
5. On the toolbar, select a **Poll Interval** from the dropdown menu.
6. On the toolbar, you can click **Clear Counters** to reset the IP statistics counters.
7. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

Use the data in the following table to help you understand miscellaneous port statistics.

Variable	Value
NoResourcesPktsDropped	The number of packets dropped due to switch memory shortage.

Comments? infodev@avaya.com

Chapter 8: Network monitoring configuration using Enterprise Device Manager

This section describes how to configure the Enterprise Device Manager (EDM) commands used to monitor your network. Use these commands to view the CPU and memory utilization, check the system log, and configure port mirroring.

Displaying CPU use with EDM

Use this procedure to view what resources you have available when using, or planning to use, resource-intensive functions. Additionally, this procedure can be used to troubleshoot slow responses.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Chassis**
4. In the Chassis work area, click the **CPU/Memory utilization** tab.
5. The variables with asterixes indicate CPU usage.

Variable definitions

The following table describes the fields on the **CPU/Mem Utilization**tab

Field	Description
Unit	Indicates the unit.
Last10Seconds	Indicates the CPU usage, in percentage, for the last 10 seconds.

Field	Description
Last1Minute	Indicates the CPU usage, in percentage, for the last minute.
Last10Minutes	Indicates the CPU usage, in percentage, for the last 10 minutes.
Last1Hour	Indicates the CPU usage, in percentage, for the last hour.
Last24Hours	Indicates the CPU usage, in percentage, for the last 24 hours.
TotalCPUUsage	Indicates the memory usage in megabytes.

Displaying memory use with EDM

Use this procedure to view how much memory you have available when using, or planning to use, resource-intensive functions. Additionally, this procedure can be used to troubleshoot slow responses or system hang-ups.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Chassis**
4. In the Chassis work area, click the **CPU/Memory utilization** tab.
5. The variables without asterixes indicate memory use.

Variable definitions

The following table describes the fields on the **CPU/Mem Utilization** tab.

Field	Description
MemoryTotalMB	Indicates the total memory present, in megabytes, on the unit.
MemoryAvailableMB	Indicates the remaining memory on the unit.

Displaying system logs using EDM

Use this procedure to display system log information. System logs record details such as the slot from which the message originated, the time the message appeared in the system log and the message description.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Diagnostics**.
3. In the Diagnostics tree, click **System Log**.
4. In the work area, click the **System Logs** tab.

Variable definitions

Use the data in the following table to help you understand the system log display.

Variable	Value
OrigUnitNumber	Indicates the slot or unit number of the originator of a log message.
MsgTime	Indicates the time (in one hundredths of a second) between system initialization and the appearance of a log message in the system log.
MsgIndex	Indicates a sequential number the system assigns to a log message when it enters the system log.
MsgSrc	Indicates if a log message loaded from non-volatile memory at system initialization or if the message generated since system initialization.
MsgString	Indicates the log message originator and the message description.

Configuring the system log using EDM

Use the following procedure to configure the system log for the purpose of troubleshooting and general monitoring.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Diagnostics**.
3. In the Diagnostics tree, click **System Log**
4. In the System Log work area, click the **System Log Settings** tab, configure the required parameters.

Variable definitions

The following table describes the fields in the **System Log Settings**tab.

Field	Description
Operation	Turns the system log on or off.
BufferFullAction	Specifies whether the system log overwrites itself or discontinues the storage of messages when the buffer is full.
Volatile - CurSize	Shows the current number of messages stored in volatile memory.
Volatile - SaveTargets	Indicates the severity of system messages to save. Available options are: • critical • critical/serious • critical/serious/inform • none Default value is critical/serious/inform.
non - Volatile - CurSize	Shows the current number of messages stored in nonvolatile memory.
non-Volatile - SaveTargets	Indicates the severity of system messages to save. Available options are:

Field	Description
	• critical • critical/serious • none Default value is critical/serious.
ClearMessageBuffers	Selects the sections of the system log to delete. Available options are: • volCritical • volSerious • volInformational • nonVolCritical • nonVolSerious

Configuring remote system logging using EDM

Use this procedure to configure and manage the logging of system messages on a secondary remote server.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Diagnostics**.
3. In the Diagnostics tree, click **System Log**.
4. In the System Log work area, click the **Remote System Log** tab.
5. In the **RemoteSyslogAddressType** section, click the type of IP Address of the remote system log server.
6. In the **RemoteSyslogAddress** box, type the IP address of the remote system log server.
7. In the **SecondarySyslogAddressType** section, click the type of IP address of the remote system log server.
8. In the **SecondarySyslogAddressType** box, type the IP address of the remote system log server.
9. Click the **Enabled** box to enable remote system logging.

   OR

   Click the **Enabled** box to disable remote system logging.

10. In the **SaveTargets** section, click the type of system messages.

Variable definitions

Use the data in the following table to help you configure the remote system log.

Field	Description
RemoteSyslogAddressType	Specifies the type of IP address of the remote system log server.
RemoteSyslogAddress	Specifies the IP address of the remote system log server to which to send system log messages.
SecondarySyslogAddressType	Specifies the type of IP address of the secondary remote system log server.
SecondarySyslogAddress	Specifies the IP address of the secondary remote system log server
Enabled	Enables or disables the remote logging of system messages.
SaveTargets	Specifies the type of system messages to send to the remote system log server. • critical—only messages classified as critical are sent to the remote system log server • critical/serious—only messages classified as critical and serious are sent to the remote system log server • critical/serious/inform—only messages classified as critical, serious, and informational are sent to the remote system log server • none—no system log messages are sent to the remote system log server

Displaying port mirroring using EDM

Use this procedure to view port mirroring settings.

Comments? infodev@avaya.com

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Diagnostics**.
3. In the Diagnostics tree, click **Port Mirrors**

Variable definitions

The following describes the variables associated with port mirroring.

Variable	Value
Instance	Numerical assignment of the port mirroring.
PortMode	The port monitoring mode. The following options are available: • Adst • Asrc • AsrcBdst • AsrcBdstorBsrcAdst • AsrcorAdst • Xrx • XrxorXtx • XrxorYtx • XrxYtx • XrxYtxOrYrxXtx • Xtx
MonitorPort	The port that is the monitoring port.
PortListX	Ports monitored for XrX/Xtx, and manytoOne related mode.
PortListY	Ports monitored for Yrx/Ytx related mode.
MacAddressA	MAC address of the monitored port using Sarc/Adst related mode.
MacAddressB	MAC address of the monitored port using Bsrc/Bdst related mode.

Configuring port mirroring using EDM

Use the following procedure to configure port mirroring.

About this task

Port mirroring allows you to monitor network traffic by sending a copy of network packets from one switch port (or entire VLAN) to a monitoring connection on a separate switch port.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Diagnostics**.
3. In the Diagnostics tree, click **Port Mirrors**
4. On the toolbar, click **Insert** .
   EDM displays the Insert Port Mirrors window.
5. Configure the parameters as required.
6. Click **Insert**.

Variable definitions

The following table describes the fields associated with the Port Mirrors tab.

Variable	Value
Instance	Numerical assignment of the port mirroring.
Port Mode	The port monitoring mode. The following options are available: • Adst • Asrc • AsrcBdst • AsrcBdstorBsrcAdst • AsrcorAdst • Xrx • XrxorXtx • XrxorYtx • XrxYtx

Comments? infodev@avaya.com

Variable	Value
	• XrxYtxOrYrxXtx • Xtx The default vale is Adst.
Monitor Port	The port that is the monitoring port.
PortListX	Ports monitored for XrX/Xtx, and manytoOne related mode.
PortListY	Ports monitored for Yrx/Ytx related mode.
MacAddressA	MAC address of the monitored port using Sarc/Adst related mode.
MacAddressB	MAC address of the monitored port using Bsrc/Bdst related mode.
Allow traffic	Allows or disallow traffic.

Displaying pluggable ports using EDM

Use this procedure to display pluggable port information on the switch.

Procedure

1. From the navigation tree, click **Edit**.
2. In the Edit tree, click **Chassis**.
3. In the Chassis tree, click **Switch/Stack**.
4. In the Switch/Stack work area, click the **Stack Info** tab to display the current stack information.
5. To select a switch unit for which to display information, click a switch row to highlight it.
6. On the toolbar, click **Pluggable Ports**.

Variable definitions

Use the data in the following table to help you understand the pluggable ports display.

Variable	Value
Unit	Identifies the unit number.
Port	Identifies the number of the pluggable port.
PortType	Identifies the type of the pluggable port.
VendorName	Identifies the vendor's name.
VendorOUI	Identifies the Vendor Organizationally Unique Identifier.
VendorPartNo	Identifies the vendor's part number.
VendorRevision	Identifies the vendor's revision.
VendorSerial	Identifies the vendor's serial number.
HWOptions	Identifies the hardware options.
DateCode	Identifies the date code.
VendorData	Identifies vendor data.
OrderCode	Identifies the order code.

Displaying TCP information for the switch using EDM

Use the following procedure to display Transmission Control Protocol (TCP) information for the switch.

Procedure

1. From the navigation tree, double-click **IP**.
2. In the IP tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **TCP Globals** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Variable definitions

The following table describes the variables associated with TCP global properties.

Variable	Value
RtoAlgorithm	Specifies the algorithm used to determine the timeout value used for retransmitting unacknowledged octets.
RtoMin	Specifies the minimum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.
RtoMax	Specifies the maximum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.
MaxConn	Specifies the limit on the total number of TCP connections that the entity can support. In entities where the maximum number of connections is dynamic, this object contains the value –1.

Displaying TCP connections using EDM

Use the following procedure to display information about the current TCP connections that the switch maintains.

Procedure

1. From the navigation tree, double-click **IP**.
2. In the IP tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **TCP Connections** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Variable definitions

The following table describes the variables associated with TCP connections.

Variable	Value
LocalAddressType	Specifies the local IP address type for this TCP connection.
LocalAddress	Specifies the local IP address for this TCP connection. In case of a connection in the listen state, which is willing to accept connections for any IP interface associated with the node, the value 0.0.0.0 is used.
LocalPort	Specifies the local port number for this TCP connection.
RemAddressType	Specifies the remote IP address type for this TCP connection.
RemAddress	Specifies the remote IP address for this TCP connection.
RemPort	Specifies the remote port number for this TCP connection.
State	Specifies the state of this TCP connection.

Displaying TCP Listeners using EDM

Use the following procedure to display information about the current TCP listeners on the switch.

Procedure

1. From the navigation tree, double-click **IP**.
2. In the IP tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **TCP Listeners** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Comments? infodev@avaya.com

Variable definitions

The following table describes the variables associated with TCP listeners.

Variable	Value
LocalAddressType	Indicates the local IP address type. Values include IPv4 or IPv6.
LocalAddress	Specifies the local IP address of the TCP listener. The value of this field can be represented in three possible ways, depending on the characteristics of the listening application: • For an application willing to accept both IPv4 and IPv6 datagrams, the value of this object is a zero-length octet string, and the value of the corresponding LocalAddressType field is unknown. • For an application willing to accept either IPv4 or IPv6 datagrams, the value of this object must be 0.0.0.0 or ::, with the LocalAddressType identifying the supported address type. • For an application that is listening for data destined only to a specific IP address, the value of this object is the specific local address, with LocalAddressType identifying the supported address type.
LocalPort	Specifies the local port number for this TCP connection.

Displaying UDP endpoints using EDM

Use the following procedure to display information about the UDP endpoints currently maintained by the switch.

Procedure

1. From the navigation tree, double-click **IP**.
2. In the IP tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **UDP Endpoints** tab.

4. On the toolbar, you can click **Refresh** to update the information.

Variable definitions

The following table describes the variables associated with UDP endpoints.

Variable	Value
LocalAddressType	Indicates the local address.
LocalAddress	Indicates the local IP address. In the case of a UDP listener that accepts datagrams for any IP interface associated with the node, the value 0.0.0.0 is used. The value of this field can be represented in three possible ways: • For an application willing to accept both IPv4 and IPv6 datagrams, the value of this object is a zero-length octet string, and the value of the corresponding LocalAddressType field is unknown. • For an application willing to accept either IPv4 or IPv6 datagrams, the value of this object must be 0.0.0.0 or ::, with the LocalAddressType identifying the supported address type. • For an application that is listening for data destined only to a specific IP address, the value of this object is the address for which this node is receiving packets, with LocalAddressType identifying the supported address type.
LocalPort	Indicates the local port.
RemoteAddressType	Indicates the remote address type.
RemoteAddress	Displays the remote IP address for this UDP endpoint. If datagrams from all remote systems are to be accepted, this value is a zero-length octet string. Otherwise, the address of the remote system from which datagrams are to be accepted (or to which all datagrams are to be sent) is displayed with the RemoteAddressType identifying the supported address type.

Variable	Value
RemotePort	Indicates the remote port. If datagrams from all remote systems are to be accepted, this value is zero.
Instance	Distinguishes between multiple processes connected to the same UDP endpoint.
Process	Displays the ID for the UDP process.

Graphing IPv6 interface ICMP statistics using EDM

Use the following procedure to display and graph the IPv6 ICMP statistics.

Procedure

1. From the navigation tree, double-click **IPv6**.
2. In the IPv6 tree, click **IPv6**.
3. In the IPv6 work area, click the **ICMP Stats** tab.
4. In the ICMP Stats section, click **Clear Counters** to reset statistics.
5. Select a Poll Interval from the drop down menu.
6. Highlight a data column to graph.
7. On the toolbar, click **Line Chart, Area Chart, Bar Chart,** or **Pie Chart**.

Variable definitions

The following table describes the variables associated with graphing IPv6 interface ICMP statistics.

Variable	Value
InMsgs	Indicates the number of ICMP messages received.
InErrors	Indicates the number of ICMP error messages received.
OutMsgs	Indicates the number of ICMP messages sent.
OutErrors	Indicates the number of ICMP error messages sent.

Displaying ICMP message statistics using EDM

Use the following procedure to display the IPv6 interface ICMP message statistics.

Procedure

1. From the navigation tree, double-click **IPv6**.
2. In the IPv6 tree, click **IPv6**.
3. In the IPv6 work area, click the **ICMP Msg Stats** tab.
4. On the toolbar, you can click **Refresh** to update the ICMP message statistics.

Variable definitions

The following table describes the variables associated with ICMP message statistics.

Variable	Value
Type	Indicates the type of packet received or sent.
InPkts	Indicates the number of packets received.
OutPkts	Indicates the number of packets sent.

Displaying IPv6 TCP global properties using EDM

Use the following procedure to display IPv6 TCP global properties.

Procedure

1. From the navigation tree, double-click **IPv6**.
2. In the IPv6 tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **TCP Globals** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Variable definitions

The following table describes the variables associated with TCP global properties.

Variable	Value
RtoAlgorithm	Specifies the algorithm used to determine the timeout value used for retransmitting unacknowledged octets.
RtoMin	Specifies the minimum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.
RtoMax	Specifies the maximum value permitted by a TCP implementation for the retransmission timeout, measured in milliseconds.
MaxConn	Specifies the limit on the total number of TCP connections that the entity can support. In entities where the maximum number of connections is dynamic, this object contains the value –1.

Displaying IPv6 TCP connections using EDM

Use the following procedure to display IPv6 TCP connections.

Procedure

1. From the navigation tree, double-click **IPv6**.
2. In the IPv6 tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **TCP Connections** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Variable definitions

The following table describes the variables associated with IPv6 TCP connections.

Variable	Value
LocalAddressType	Specifies the local IP address type for this TCP connection.
LocalAddress	Specifies the local IP address for this TCP connection. In case of a connection in the listen state, which is willing to accept connections for any IP interface associated with the node, the value 0.0.0.0 is used.
LocalPort	Specifies the local port number for this TCP connection.
RemAddressType	Specifies the remote IP address type for this TCP connection.
RemAddress	Specifies the remote IP address for this TCP connection.
RemPort	Specifies the remote port number for this TCP connection.
State	Specifies the state of this TCP connection.

Displaying IPv6 TCP listeners using EDM

Use the following procedure to display IPv6 TCP listeners.

Procedure

1. From the navigation tree, double-click **IPv6**.
2. In the IPv6 tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **TCP Listeners** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Comments? infodev@avaya.com

Variable definitions

The following table describes the variables associated with IPv6 TCP listeners.

Variable	Value
LocalAddressType	Indicates the local IP address type. Values include IPv4 or IPv6.
LocalAddress	Indicates the local IPv4 or IPv6 address.
LocalPort	Indicates the local port.

Displaying IPv6 UDP endpoints using EDM

Use the following procedure to display IPv6 UDP endpoints.

Procedure

1. From the navigation tree, double-click **IPv6**.
2. In the IPv6 tree, click **TCP/UDP**.
3. In the TCP/UDP work area, click the **UDP Endpoints** tab.
4. On the toolbar, you can click **Refresh** to update the information.

Variable definitions

The following table describes the variables associated with iPv6 UDP endpoints.

Variable	Value
LocalAddressType	Indicates the local address.
LocalAddress	Indicates the local IP address. In the case of a UDP listener that accepts datagrams for any IP interface associated with the node, the value 0.0.0.0 is used. The value of this field can be represented in three possible ways: • For an application willing to accept both IPv4 and IPv6 datagrams, the value of this object is a zero-length octet string, and the

Variable	Value
	value of the corresponding LocalAddressType field is unknown. • For an application willing to accept either IPv4 or IPv6 datagrams, the value of this object must be 0.0.0.0 or ::, with the LocalAddressType identifying the supported address type. • For an application that is listening for data destined only to a specific IP address, the value of this object is the address for which this node is receiving packets, with LocalAddressType identifying the supported address type.
LocalPort	Indicates the local port.
RemoteAddressType	Indicates the remote address type.
RemoteAddress	Displays the remote IP address for this UDP endpoint. If datagrams from all remote systems are to be accepted, this value is a zero-length octet string. Otherwise, the address of the remote system from which datagrams are to be accepted (or to which all datagrams are to be sent) is displayed with the RemoteAddressType identifying the supported address type.
RemotePort	Indicates the remote port. If datagrams from all remote systems are to be accepted, this value is zero.
Instance	Distinguishes between multiple processes connected to the same UDP endpoint.
Process	Displays the ID for the UDP process.

Displaying network topology information using EDM

Use this procedure to display network topology information.

Procedure

1. From the navigation tree, double-click **Edit**.
2. In the Edit tree, double-click **Diagnostics**.

Comments? infodev@avaya.com

3. In the Diagnostics tree, click **Topology**.
4. In the Topology work area, click the **Topology** tab.
5. In the **Status** section, select a radio button.
6. On the toolbar, click **Apply**.

Variable definitions

The following table describes the variables associated with network topology.

Variable	Value
IpAddr	Indicates the IP address of the device.
Status	Specifies whether Avaya topology is on (topOn) or off (topOff) for the device. The default value is topOn.
NmmLstChg	Indicates the value of sysUpTime when an entry in the network management MIB (NMM) topology table is added, deleted, or modified. If the table did not change since the last cold or warm start of the agent, the value is zero. This is a read-only cell.
NmmMaxNum	Indicates the maximum number of entries in the NMM topology table.
NmmCurNum	Indicates the current number of entries in the NMM topology table.

Displaying the topology table using EDM

Use this procedure to display the topology table.

Procedure

1. From the navigation tree, double-click **Edit**.
2. In the Edit tree, double-click **Diagnostics**.
3. In the Diagnostics tree, click **Topology**.
4. In the Topology work area, click the **Topology Table** tab.

Variable definitions

The following table describes the variables associated with the topology table.

Variable	Value
Slot	Indicates the slot number in the chassis that the topology message is received.
Port	Indicates the port that the topology message is received.
IpAddr	Indicates the IP address of the sender of the topology message.
SegId (Slot/Port)	Indicates the segment identifier of the segment from which the remote agent sent the topology message. This value is extracted from the message.
MacAddr	Indicates the MAC address of the sender of the topology message.
ChassisType	Indicates the chassis type of the device that sent the topology message.
BkplType	Indicates the backplane type of the device that sent the topology message.
LocalSeg	Indicates if the sender of the topology message is on the same Ethernet segment as the reporting agent.
CurState	Indicates the current state of the sender of the topology message. The choices are: • topChanged - Topology information recently changed. • heartbeat — Topology information is unchanged. • new — The sending agent is a new state.

Using the EDM MIB Web page for SNMP Get and Get-Next

You can use the EDM Management Information Base (MIB) Web page to view the response of an SNMP Get and Get-Next request for any Object Identifier (OID).

Procedure

1. From the navigation tree, click **Administration**.
2. In the Administration tree, click **MIB Web Page**.
3. In the **MIB Name/OID** box, enter the object name or OID.
4. Click **Get**.
   EDM displays the result of the request in the Result area of the window. EDM displays an error description if the request is unsuccessful.
5. Click **Get Next** to retrieve the information of the next object in the MIB.
6. Repeat step 3 as required.

Using the EDM MIB Web page for SNMP walk

You can use SNMP walk to retrieve a MIB subtree of the root SNMP object.

Perform this procedure to request the result of MIB Walk.

Procedure

1. From the navigation tree, click **Administration**.
2. In the Administration tree, click **MIB Web Page**.
3. In the **MIB Name/OID** box, enter the object name or OID.
4. Click **Walk** .
   EDM displays the result of the request in the Result area. EDM displays an error description if the request is unsuccessful.

Comments? infodev@avaya.com

Chapter 9: RMON configuration using Enterprise Device Manager

This section describes the Enterprise Device Manager (EDM) commands used to configure and manage RMON.

Managing RMON alarms using EDM

Use the procedures in this section to display, create, configure, and delete RMON alarms.

You can use RMON alarms to be notified when the values of a variable go out of range. All RMON alarms share the following characteristics:

- An upper and lower threshold value is defined.
- A corresponding rising and falling event occurs.
- An alarm interval or polling period is reached.

Once an alarm is activated, you can view the activity in a log or a trap log.

Displaying RMON alarms using EDM

Use the following procedure to view the configured RMON alarms.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click **Alarms** tab.

Variable definitions

The following table describes the variables associated with RMON alarms.

Variable	Value
Index	Uniquely identifies an entry in the alarm table. Each entry defines a diagnostic sample at a particular interval for an object on the device.
Interval	Indicates the time period (in seconds) over which the data is sampled and compared with the rising and falling thresholds.
Variable	Insert a description of this field. Indicates the name and type of alarm. *alarmname.x* Here x=0 indicates a chassis alarm. *alarmname*. where the user must specify the index. This index is a card number for module-related alarms, an STG ID for spanning tree group alarms (the default STG is 1, other STG IDs are userconfigured), or the Ether Statistics Control Index for RMON Stats alarms *alarmname* with no dot or index is a port-related alarm and results in display of the port selection tool.
SampleType	Indicates the sample type. Available options are: • absoluteValue • deltaValue
Value	Indicates the value of the statistic during the last sampling period. For example, if the sample type is deltaValue, this value is the difference between the samples at the beginning and end of the period. If the sample type is absoluteValue, this value is the sampled value at the end of the period. This value is compared with the rising and falling thresholds. The value during the current sampling period is not made available until the period is completed and remains available until the next period is completed.
StartupAlarm	Indicates the alarm that might be sent after this entry is first set to Valid. If the first sample

Variable	Value
	after this entry becomes valid is greater than or equal to the risingThreshold and alarmStartupAlarm is equal to risingAlarm(1) or risingOrFallingAlarm(3), a single rising alarm is generated. If the first sample after this entry becomes valid is less than or equal to the fallingThreshold and alarmStartupAlarm is equal to fallingAlarm(2) or risingOrFallingAlarm(3) a single falling alarm is generated.
RisingThreshold	Generates a single event if the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is less than this threshold.
RisingEventIndex	Indicates the index of the event entry that is used after a rising threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)
FallingThreshold	Generates a single event if the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is greater than this threshold.
FallingEventIndex	Indicates the index of the event entry that is used after a falling threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)
Owner	Indicates the network management system that created this entry.
Status	Indicates the status of this alarm entry.

Creating a RMON alarm using EDM

Use the following procedure to create an alarm to receive statistics and history using default values.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click **Alarms** tab.
5. In the toolbar, click **Insert**.
   EDM displays the Insert Alarm window.
6. Configure the parameters as require for the alarm.
7. Click **Insert**.

Variable definitions

The following table describes the fields of Insert Alarm dialog box.

Field	Description
Variable	Indicates the name and type of alarm. *alarmname.x* Here x=0 indicates a chassis alarm. *alarmname*. where the user must specify the index. This index is a card number for module-related alarms, an STG ID for spanning tree group alarms (the default STG is 1, other STG IDs are userconfigured), or the Ether Statistics Control Index for RMON Stats alarms *alarmname* with no dot or index is a port-related alarm and results in display of the port selection tool.
SampleType	Indicates the sample type. Available options are: • absoluteValue • deltaValue
Interval	Indicates the time period (in seconds) over which the data is sampled and compared with the rising and falling thresholds.
Index	Uniquely identifies an entry in the alarm table. Each such entry defines a diagnostic sample at a particular interval for an object on the device.

Field	Description
RisingThreshold	Generates a single event if the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is less than this threshold.
RisingEventIndex	Indicates the index of the event entry that is used after a rising threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)
FallingEventIndex	Indicates the index of the event entry that is used after a falling threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)
Owner	Indicates the network management system that created this entry.

Configuring RMON alarms using EDM

Use the following procedure to configure an alarm to receive statistics and history using default values. Configuring RMON alarms is an integral part of proactive management. An alarm is triggered once the value of the variable falls outside the allowable operational range.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click **Alarms** tab.
5. In the toolbar, click **Insert**.
   EDM displays the insert Alarm window.
6. Configure the parameters as required for the alarm.
7. Click **Insert**.

Variable definitions

The following table describes the fields of the Insert Alarm dialog box.

Field	Description
Index	Uniquely identifies an entry in the alarm table. Each such entry defines a diagnostic sample at a particular interval for an object on the device.
Interval	Indicates the time period (in seconds) over which the data is sampled and compared with the rising and falling thresholds.
Variable	Indicates the name and type of alarm. `alarmname.x` Here x=0 indicates a chassis alarm. `alarmname`. where the user must specify the index. This index is a card number for module-related alarms, an STG ID for spanning tree group alarms (the default STG is 1, other STG IDs are userconfigured), or the Ether Statistics Control Index for RMON Stats alarms `alarmname` with no dot or index is a port-related alarm and results in display of the port selection tool.
Sample Type	Indicates the sample type. Available options are: • absoluteValue • deltaValue
Value	Indicates the value of the statistic during the last sampling period. For example, if the sample type is deltaValue, this value is the difference between the samples at the beginning and end of the period. If the sample type is absoluteValue, this value is the sampled value at the end of the period. This value is compared with the rising and falling thresholds. The value during the current sampling period is not made available until the period is completed and remains available until the next period is completed.
StartupAlarm	Indicates the alarm that might be sent after this entry is first set to Valid. If the first sample after this entry becomes valid is greater than

Field	Description
	or equal to the risingThreshold and alarmStartupAlarm is equal to risingAlarm(1) or risingOrFallingAlarm(3), a single rising alarm is generated. If the first sample after this entry becomes valid is less than or equal to the fallingThreshold and alarmStartupAlarm is equal to fallingAlarm(2) or risingOrFallingAlarm(3) a single falling alarm is generated.
RisingThreshold	Generates a single event if the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is less than this threshold.
RisingEventIndex	Indicates the index of the event entry that is used after a rising threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)
FallingThreshold	Generates a single event if the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is greater than this threshold.
FallingEventIndex	Indicates the index of the event entry that is used after a falling threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)
Owner	Indicates the network management system that created this entry.
Status	Indicates the status of this alarm entry.

Deleting a RMON alarm using EDM

Use the following procedure to delete a previously created alarm.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.

3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click the **Alarms** tab.
5. Select the alarm you want to delete.
6. In the toolbar, click **Delete**.

Variable definitions

The following table describes the fields on the Alarms tab.

Field	Description
Index	Uniquely identifies an entry in the alarm table. Each such entry defines a diagnostic sample at a particular interval for an object on the device.
Interval	Indicates the interval in seconds over which data is sampled and compared with the rising and falling thresholds.
Variable	Indicates the object identifier of the particular variable to be sampled. Only variables that resolve to an ASN.1 primitive type of INTEGER (INTEGER, Counter, Gauge, or TimeTicks) can be sampled.
Sample Type	Indicates the method of sampling the selected variable and calculating the value to be compared against the thresholds. If the value of this object is absoluteValue(1), the value of the selected variable is compared directly with the thresholds at the end of the sampling interval. If the value of this object is deltaValue(2), the value of the selected variable at the last sample is subtracted from the current value, and the difference compared with the thresholds.
Value	Indicates the value of the statistic during the last sampling period. For example, if the sample type is deltaValue, this value is the difference between the samples at the beginning and end of the period. If the sample type is absoluteValue, this value is the sampled value at the end of the period. This value is compared with the rising and falling thresholds. The value during the current sampling period is not made

Field	Description
	available until the period is completed and remains available until the next period is completed.
StartupAlarm	Indicates the alarm that might be sent after this entry is first set to Valid. If the first sample after this entry becomes valid is greater than or equal to the risingThreshold and alarmStartupAlarm is equal to risingAlarm(1) or risingOrFallingAlarm(3), a single rising alarm is generated. If the first sample after this entry becomes valid is less than or equal to the fallingThreshold and alarmStartupAlarm is equal to fallingAlarm(2) or risingOrFallingAlarm(3) a single falling alarm is generated.
Rising Threshold	Indicates the threshold for the sampled statistic. After the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is less than this threshold, a single event is generated. A single event is also generated if the first sample after this entry becomes valid is greater than or equal to this threshold and the associated alarmStartupAlarm is equal to risingAlarm(1) or risingOrFallingAlarm(3). After a rising event is generated, another such event is not generated until the sampled value falls below this threshold and reaches the alarmFallingThreshold.
RisingEventIndex	Indicates the index of the eventEntry that is used after a rising threshold is crossed. The eventEntry identified by a particular value of this index is the same as identified by the same value of the eventIndex object. If there is no corresponding entry in the eventTable no association exists. In particular, if this value is zero, no associated event is generated, because zero is not a valid event index.
Falling Threshold	Indicates the threshold for the sampled statistic. After the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is greater than this threshold, a single event is generated. A single event is also generated if the first sample after this entry becomes

Field	Description
	valid is less than or equal to this threshold and the associated alarmStartupAlarm is equal to fallingAlarm(2) or risingOrFallingAlarm(3). After a falling event is generated, another such event is not generated until the sampled value rises above this threshold and reaches the alarmRisingThreshold.
FallingEventIndex	Indicates the index of the eventEntry that is used after a falling threshold is crossed. The eventEntry identified by a particular value of this index is the same as identified by the same value of the eventIndex object. If there is no corresponding entry in the eventTable no association exists. In particular, if this value is zero, no associated event is generated, because zero is not a valid event index.
Owner	Indicates the network management system that created this entry.
Status	Indicates the status of this alarm entry.

Managing log information using EDM

Use the following procedure to chronicle and describe alarm activity.

Procedure

1. From the navigation tree, double-click **Servicebility**.
2. In the Servicebility tree, double-click **RMON**.
3. In the RMON tree, click **Alarms**.
4. In the Alarms work area, click **Log**.

Comments? infodev@avaya.com

Variable definitions

The following table describes the variables associated with the RMON alarms log tab.

Variable	Value
Time	Specifies when an event occurs that activates the log entry.
Description	Specifies whether the even is a rising or falling event.
EventIndex	Specifies the event index.

Managing RMON events using EDM

Use the procedures in this section to display, add, or delete RMON events.

You can use RMON events and alarms together to notify you when values in your network are outside of a specified range. When a value exceeds the specified range, the alarm is triggered. The event specifies how the activity is recorded.

Displaying RMON events using EDM

Use this procedure to view RMON events when troubleshooting or monitoring network issues.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click **Events** tab.

Variable definitions

The following table describes the variables associated with RMON events.

Variable	Value
Index	Uniquely identifies an entry in the event table. Each entry defines one event that is to be generated after the appropriate conditions occur.
Description	Specifies whether the event is a rising or falling event.
Type	Indicates the type of notification that EDM provides about this event. In the case of log, an entry is made in the log table for each event. In the case of trap, an SNMP trap is sent to one or more management stations. Possible notifications are: • none • log • trap • log-and-trap
Community	Indicates the SNMP community string acts as a password. Only those management applications with this community string can view the alarms.
LastTimeSent	Indicates the value of sysUpTime when the event entry last generated an event. If the entry did not generate events, the value is zero.
Owner	If traps are specified to be sent to the owner, this is the name of the machine receives alarm traps.

Creating a RMON event using EDM

Use this procedure to create an event.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.

Comments? infodev@avaya.com

3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click the **Events** tab.
5. In the toolbar, click **Insert**.
   EDM displays the Insert Events window.
6. Configure the parameters as required.
7. Click **Insert**.

Variable definitions

The following table describes the variables associated with creating an event.

Variable	Value
Index	Uniquely identifies an entry in the event table. Each entry defines one event that is to be generated after the appropriate conditions occur.
Description	Specifies whether the event is a rising or falling event.
Type	Indicates the type of notification that EDM provides about this event. In the case of log, an entry is made in the log table for each event. In the case of trap, an SNMP trap is sent to one or more management stations. Possible notifications are: • none • log • trap • log-and-trap
Community	Indicates the SNMP community string acts as a password. Only those management applications with this community string can view the alarms.
Owner	If traps are specified to be sent to the owner, this is the name of the machine receives alarm traps.

Deleting a RMON event using EDM

Use this procedure to delete an event.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Alarms**
4. In the Alarms work area, click the **Events** tab to view the history of the event.
5. In the table, select the event row you want to delete.
6. On the toolbar, click **Delete**.

Managing RMON history using EDM

Use the procedures in this section to manage RMON history statistics.

Displaying RMON history using EDM

Ethernet history records periodic statistical samples from a network. A sample is called a history and is gathered in time intervals referred to as "buckets."

Histories establish a time-dependent method for gathering RMON statistics on a port.

The default values for history are:

- Buckets are gathered at 30-minute intervals.
- Number of buckets gathered is 50.

the time interval and the number of buckets can be configured. After the last bucket is reached, bucket 1 is dumped and "recycled" to hold a new bucket of statistics. Then bucket 2 is dumped.

Use this procedure to display information regarding the configuration of RMON history.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.

3. In the RMON tree, click **Control**
4. In the work area, click the **History** tab.

Variable definitions

The following table describes the fields of the **History** tab.

Field	Description
Index	Indicates a unique value assigned to each interface. An index identifies an entry in a table.
Port	Indicates an Ethernet interface on the device.
BucketsRequested	Indicates the requested number of discrete time intervals data is saved in the part of the media-specific table associated with this entry.
BucketsGranted	Indicates the number of discrete sampling intervals data is saved in the part of the media-specific table associated with this entry. If instances after the actual number of buckets associated with this entry is less than the value of this object, at the end of a sampling interval a new bucket is added to the media-specific table.
Interval	Indicates the interval in seconds that the data is sampled for each bucket in the part of the media-specific table associated with this entry. You can set this interval to a number of seconds between 1 and 3600 (1 hour). Because the counters in a bucket might overflow at their maximum value with no indication, note the possibility of overflow in the associated counters. Consider the minimum time in which a counter could overflow on a particular media type and set the historyControlInterval object to a value less than this interval. This interval is typically most important for the octets counter in a media-specific table. For example, on an Ethernet network, the etherHistoryOctets counter could overflow in about one hour at the Ethernet's maximum utilization.

Field	Description
Owner	Indicates the network management system that created this entry.

Creating a history using EDM

RMON can be used to collect statistics at intervals. For example, if switch performance is monitored over a weekend, enough buckets to cover two days must be set aside. To do this, set the history to gather one bucket each hour, thus covering a 48-hour period. After history characteristics are set, they cannot be modified; the history must be deleted and another created.

Use the following procedure to establish a history for a port, and set the bucket interval.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Control**.
4. In the Control work area, click the**History** tab.
5. In the toolbar, click **Insert**.
   EDM displays the Insert History window.
6. In the fields provided, enter the information for the new RMON history.
7. Click **Insert**.

Variable definitions

The following table describes the variables associated with the RMON history table.

Variable	Value
Index	Indicates a unique value assigned to each interface. An index identifies an entry in a table.
Port	Indicates any Ethernet interface on the device.
BucketsRequested	Indicates the requested number of discrete time intervals over which data is to be saved in the part of the media-specific table associated with this entry.

Variable	Value
Interval	Indicates the interval in seconds over which the data is sampled for each bucket in the part of the media-specific table associated with this entry. You can set this interval to a number of seconds between 1 and 3600 (1 hour). Because the counters in a bucket might overflow at their maximum value with no indication, note the possibility of overflow in the associated counters. Consider the minimum time in which a counter could overflow on a particular media type and set the historyControlInterval object to a value less than this interval. This interval is typically most important for the octets counter in a mediaspecific table. For example, on an Ethernet network, the etherHistoryOctets counter could overflow in about one hour at the Ethernet's maximum utilization.
Owner	Indicates the network management system that created this entry.

Deleting RMON history table entries using EDM

Use this procedure to delete an entry in the RMON history table.

Procedure

1. From the navigation tree, click **Serviceability**.
2. From the Serviceability tree, click **RMON**.
3. From the RMON tree, click **Control**.
4. In the Control work area, click the **History** tab.
5. Highlight the entry to be deleted, click **Delete** on the toolbar.

Variable definitions

The following table describes the variables associated with the RMON history table.

Variable	Value
Index	Indicates a unique value assigned to each interface. An index identifies an entry in a table.
Port	Indicates any Ethernet interface on the device.
BucketsRequested	Indicates the requested number of discrete time intervals over which data is to be saved in the part of the media-specific table associated with this entry.
BucketsGranted	Indicates the granted number of discrete time intervals over which data is to be saved in the part of the media-specific table associated with this entry.
Interval	Indicates the interval in seconds over which the data is sampled for each bucket in the part of the media-specific table associated with this entry. You can set this interval to a number of seconds between 1 and 3600 (1 hour). Because the counters in a bucket might overflow at their maximum value with no indication, note the possibility of overflow in the associated counters. Consider the minimum time in which a counter could overflow on a particular media type and set the historyControlInterval object to a value less than this interval. This interval is typically most important for the octets counter in a mediaspecific table. For example, on an Ethernet network, the etherHistoryOctets counter could overflow in about one hour at the Ethernet's maximum utilization.
Owner	Indicates the network management system that created this entry.

Managing RMON statistics using EDM

Use the procedures in this section to manage RMON Ethernet statistics.

Comments? infodev@avaya.com

The RMON agent continuously collects statistics and monitors switch performance. You can view these statistics. The statistics are available in two formats:

- Cumulative statistics for all Ethernet interfaces.
- Statistics for individual Ethernet interfaces.

Displaying RMON statistics using EDM

EDM gathers Ethernet statistics that can be graphed in a variety of formats or saved to a file that can be exported to an outside presentation or graphing application.

Use the following procedure to display the RMON Ethernet statistics.

Procedure

1. From the navigation tree, double-click **Servicability**.
2. In the Serviceability tree, double-click **RMON**.
3. In the RMON tree, click **Control**.
4. In the work area, click the **Ether Stats** tab to view RMON ethernet statistics.

Variable definitions

The following table describes the fields on the **Ether Stats** tab.

Field	Description
Octets	Indicates the total number of octets of data (including those in bad packets) received on the network (excluding framing bits but including FCS octets). You can use this object as a reasonable estimate of Ethernet utilization. For greater precision, sample the etherStatsPkts and etherStatsOctets objects before and after a common interval.
Pkts	Indicates the total number of packets (including bad packets, broadcast packets, and multicast packets) received.
BroadcastPkts	Indicates the total number of good packets received that were directed to the broadcast address. This does not include multicast packets.
MulticastPkts	Indicates the total number of good packets received that were directed to a multicast

Field	Description
	address. This number does not include packets directed to the broadcast address.
CRCAlignErrors	Indicates the total number of packets received that had a length (excluding framing bits, but including FCS octets) of between 64 and 1518 octets, inclusive, but had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error).
UndersizePkts	Indicates the total number of packets received that were less than 64 octets long (excluding framing bits but including FCS octets) and were otherwise well formed.
OversizePkts (>1518)	Indicates the total number of packets received that were longer than 1518 octets (excluding framing bits but including FCS octets) and were otherwise well formed.
Fragments	Indicates the total number of packets received that were less than 64 octets in length (excluding framing bits but including FCS octets) and had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error). Normal behavior for etherStatsFragments is to increment because it counts both runts (which are normal occurrences due to collisions) and noise hits.
Collisions	Indicates the best estimate of the total number of collisions on this Ethernet segment.
Jabbers	Indicates the total number of packets received that were longer than 1518 octets (excluding framing bits, but including FCS octets), and had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error). Jabber is defined as the condition where a packet exceeds 20 ms. The allowed range to detect jabber is between 20 ms and 150 ms.

Comments? infodev@avaya.com

Field	Description
1..64	Indicates the total number of packets (including bad packets) that were transmitted and received on this port between 1 and 64 octets in length (excluding framing bits but including FCS octets).
65..127	Indicates the total number of packets (including bad packets) that were transmitted and received on this port between 65 and 127 octets in length (excluding framing bits but including FCS octets).
128..255	Indicates the total number of packets (including bad packets) that were transmitted and received on this port between 128 and 255 octets in length (excluding framing bits but including FCS octets).
256..511	Indicates the total number of packets (including bad packets) that were transmitted and received on this port between 256 and 511 octets in length (excluding framing bits but including FCS octets).
512..1023	Indicates the total number of packets (including bad packets) that were transmitted and received on this port between 512 and 1023 octets in length (excluding framing bits but including FCS octets).
1024..1518	Indicates the total number of packets (including bad packets) that were transmitted and received on this port between 1024 and 1518 octets in length (excluding framing bits but including FCS octets).

Enabling ethernet statistics gathering using EDM

Use the following procedure to gather ethernet statistics.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Control**.
4. In the Control work area, click the **Ether Stats** tab.

5. In toolbar, click **Insert**.
   EDM displays the Insert Ether Stats window.
6. Enter the port number you want to use in the **Port** field. You can either type the port number, or click Port ellipse (...) to select a port number from the Port list.
7. Click **Insert**.

Variable definitions

The following table describes the fields of Insert Ether Stats screen.

Field	Description
Index	Indicates a unique value assigned to each interface. An index identifies an entry in a table.
Port	Indicates a port on the device.
Owner	Indicates the network management system that created this entry.

Disabling RMON ethernet statistics gathering using EDM

Use the following procedure to disable Ethernet statistics.

Procedure

1. From the navigation tree, click **Serviceability**.
2. In the Serviceability tree, click **RMON**.
3. In the RMON tree, click **Control**
4. In the Control work area, click the**Ether Stats** tab.
5. Select the row that contains the record that you want to delete.
6. In the toolbar, click **Delete**.