CHAPTER 1

# Overview

This manual describes how to configure and manage the FastHub using a standard SNMP-based network-management application. The manual also describes the standard MIB objects and MIB object extensions supported by the FastHub.

Cisco documentation and additional literature are available on a CD-ROM called Cisco Connection Documentation, Enterprise Series, which ships with your chassis. The CD is updated and shipped monthly, so it might be more up to date than printed documentation. To order additional copies of the Cisco Connection Documentation, Enterprise Series CD, contact your local sales representative or call Customer Service. The CD is available both as a single CD and as an annual subscription. You can also access Cisco technical documentation on the World Wide Web URL http://www.cisco.com.

---

**Note** The Cisco Connection Documentation, Enterprise Series CD was previously called UniverCD.

---

Using the Simple Network Management Protocol (SNMP), the FastHub communicates with the third-party network-management application through its in-band management interface (the SNMP agent). The management information used to configure and monitor a FastHub are represented as objects in a database called a Management Information Base (MIB). The FastHub can be managed in-band through any SNMP-compatible workstation or through Telnet. The FastHub supports standard MIB-II objects as well as custom extensions found in the enterprise-specific MIB. The extensions provide access to unique FastHub features and other management functions.

The complete set of FastHub MIB objects are listed by function (user action) in the "FastHub MIB Implementation" section in this chapter.

# FastHub and SNMP Management Platforms

In general, you use SNMP network-management applications to locate the FastHub icon and access the table of FastHub objects. You can then view the characteristics and counters describing the FastHub and set object values as defined in the FastHub-supported MIBs.

CiscoWorks applications, one method of SNMP network management, are integrated on several SNMP-based network management platforms, including SunNet Manager, HP Open View, and IBM NetView. Contact Cisco Systems or your authorized reseller for detailed information on CiscoWorks.

# Supported TCP/IP Protocols

The FastHub uses a subset of the Transmission Control Protocol/Internet Protocol (TCP/IP) suite as the underlying mechanism to transport the SNMP. The following protocols are implemented in the FastHub:

- Internet Protocol (IP)
- Internet Control Message Protocol (ICMP)
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
- Trivial File Transfer Protocol (TFTP)
- Bootstrap Protocol (BOOTP)
- Address Resolution Protocol (ARP)
- Dynamic Host Configuration Protocol (DHCP)
- Routing Information Protocol (RIP)
- Domain Name Service (DNS)

# Before Beginning...

Before beginning any in-band management, the FastHub must be configured for SNMP management. To do this, assign an IP address to the FastHub, using the IP Configuration Menu described in the “Out-of-Band Management” chapter of the *Catalyst FastHub 100+ Series Installation and Configuration Guide*. Also included in the guide is a procedure to assign an IP address using BOOTP/DHCP.

# Using Telnet

You can use any Telnet TCP/IP package to access the in-band interface. The FastHub supports up to seven simultaneous Telnet sessions. The Telnet TCP/IP package must support VT-100 terminal emulation.

# SNMP Agent

The network management module (NMM) SNMP agent implements SNMP version 1, specifically supporting:

- Trivial authentication using community strings with no privacy.
- Get, Get-Next, and Set operations, as well as the ability to generate traps.

## Community Strings

The SNMP agent implements two separate community strings. The first community string, the Get community string, has a default ASCII value of public and can be used by a management workstation to send Get and Get-Next requests to the agent.

The second string, the Set community string, has a default ASCII value of private and can be used in Get, Get-Next, and Set requests.

These strings are modifiable only through the out-of-band management console. If configured to do so, the SNMP agent generates authenticationFailure traps whenever it receives a request with an invalid community string.

## Set Clients

To provide additional security, the NMM SNMP agent uses “set client” IP addresses. Up to four IP addresses can be defined as set clients, giving workstations the authority to issue Set requests and add other set clients. The list of set clients is initially empty, and any set client workstation can set the first address. After the first address (or addresses) are set, only management workstations having the same IP address as those on the list can add more addresses or set other MIB objects. If a management workstation does not have the same IP address as the address on the list, Set requests are dropped (without notification).

## Trap Clients

Traps use their own community strings and receiver addresses. A trap receiver, also called a “trap client,” is a management workstation configured to receive and process traps. The method by which a trap client workstation is configured is management-platform dependent. The NMM firmware maintains a list with up to four trap IP addresses and four trap community strings, one for each possible trap client. The trap client list is shipped empty from the factory. An empty trap client list disables the generation of all traps.

The FastHub can generate the following traps:

- Five standard MIB-II traps
- Three repeater MIB traps
- Two remote monitoring (RMON) MIB traps
- Four enterprise-specific traps

The MIB object mrNetMgmtEnableAuthenTraps can be set to suppress the generation of the authenticationFailure traps.

### Configuring a Trap Client

To configure a trap client, use the following MIB objects:

#### mrNetMgmtTrapClientTable

This table contains four entries that list the management workstations that are to receive traps generated from this agent.

### mrNetMgmtTrapClientIndex

This read-only MIB object provides identification of a trap client entry.

### mrNetMgmtTrapClientName

This read-write MIB object specifies the trap client's name or IP address.

### mrNetMgmtTrapClientComm

This read-write MIB object specifies the community string used for traps sent to this trap client.

### mrNetMgmtTrapClientStatus

Setting this read-write MIB object to "invalid" invalidates the corresponding entry. That is, it disassociates the IP address or community string identified with that entry from the table. It is an implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management workstations must be prepared to receive tabular information from agents that corresponds to entries not currently in use.

## FastHub Traps

The FastHub generates the following traps:

### Enterprise-Specific Traps

### logonIntruder

A user is repeatedly trying to log on to the management console using an invalid password. You can define the number of invalid passwords permitted before this trap is generated. The FastHub can shut down the management console following the generation of this trap.

### hubStackDiagnostic

The FastHub issues this trap when its power-on self-test (POST) does not pass all tests. However, note that some failures are catastrophic, preventing the generation of this trap.

### powerSupplyFailure

This trap is issued when either the internal power supply or the redundant power supply (RPS) fails.

### ipAddressChange

This trap is issued when the NMM SNMP agent is unable to complete its DHCPDISCOVER/DHCPREQUEST process, when it fails to extend the lease for the current address, or when it accepts an address change from the user.

## Repeater Traps

### rptrHealth

This trap conveys information related to the operational status of the FastHub. This trap is sent either when the value of rptrOperStatus changes or when a nondisruptive test completes.

### rptrGroupChange

This trap is sent when a change occurs in the group structure of the FastHub. This occurs only when a group is logically or physically removed from or added to a repeater.

### rptrResetEvent

This trap conveys information related to the operational status of the FastHub. This trap is sent on completion of a reset action (such as an SNMP Set on the rptrReset object).

## Remote Network Monitoring Traps

### risingAlarm

This SNMP trap is generated when an alarm entry crosses its rising threshold and generates an event that is configured for sending SNMP traps.

### fallingAlarm

This SNMP trap is generated when an alarm entry crosses its falling threshold and generates an event that is configured for sending SNMP traps.

### MIB II Traps

### warmStart

Generated when the repeater is reset or after the completion of a firmware upgrade.

### coldStart

Generated upon a power-on reset.

### linkDown

This trap is currently not generated by the FastHub.

### linkUp

This trap is currently not generated by the FastHub.

### authenticationFailure

Generated when the FastHub receives an SNMP message that is not accompanied by a valid community string.

# Upgrading Firmware

Firmware upgrades transfer (download) an upgrade file directly into the FastHub FLASH memory.

The in-band upgrade is done through a FastHub 100BaseT port using TFTP and requires that the NMM be configured with an IP address. The out-of-band upgrade uses a serial link to the NMM's console port.

Only one upgrade attempt can be in progress at any one time. FastHub firmware returns an error indication when it detects an upgrade conflict.

**Note** The following upgrade procedures assume that the FastHub has been appropriately configured with a valid IP address (required for in-band upgrade). To assign an IP address to the FastHub, use the IP Configuration Menu, as described in the "Out-of-Band Management" chapter of the *Catalyst FastHub 100+ Series Installation and Configuration Guide*. Also included in the guide is a procedure to assign an IP address using BOOTP/DHCP.

## In-Band Upgrade

There are two ways to perform an in-band upgrade:

- FastHub-directed: The NMM SNMP agent controls the upgrade, first locating the upgrade file, then issuing the first TFTP read request.
- Workstation-directed: You control the upgrade from your management workstation, generating the TFTP write request to the FastHub.

### FastHub-Directed Upgrade

Follow these steps to upgrade the FastHub firmware:

**Step 1** Determine the size of the upgrade file to be loaded, then use mrUpgradeFlashSize to ensure that there is available FLASH memory.

**Step 2** If necessary, use the following MIB objects to obtain information about the last upgrade performed:

mrUpgradeLastUpgradeTime—displays the date and time of the last upgrade.

mrUpgradeLastUpgradeSource—displays the IP address of the TFTP server or client that sent the last upgrade file. If the last upgrade was out-of-band through the NMM console port, 0.0.0.0 is displayed.

mrUpgradeLastUpgradeStatus—displays the status of the last upgrade.

**Step 3** Use mrUpgradeTFTPServerAddress to specify the name of the TFTP server where the upgrade file is located. Note that if the first non-blank character specified is a NetASCII numeral, the name is assumed to be an IP address. If the first non-blank character is not a NetASCII numeral, it is assumed to be a fully qualified domain name server name, and the Domain Name System (DNS) protocol is used to resolve it to an IP address.

You can also (optionally) provide a name or IP address of a default gateway, as in the case of the TFTP server being located on a non-local IP network.

**Step 4** Use mrUpgradeTFTPLoadFilename to specify the name of the firmware upgrade file.

**Step 5** Use mrUpgradeTFTPInitiate to initiate the firmware upgrade.

Once initiated, the FastHub-directed method begins issuing a file-open request to the TFTP server. Read requests then follow to obtain the file content. The transfer mode used is octet, the opposite of the text transfer mode. The upgrade process either completes successfully or times out. The time-out interval is approximately 30 seconds. After a successful transfer of the upgrade file, the FastHub resets and executes the new firmware.

### Workstation-Directed Upgrade

In the workstation-directed upgrade, you need a workstation equipped with a TFTP client application. Internet-based UNIX computers such as the Sun workstation usually come configured with such an application. On DOS, these types of applications are available from a number of different vendors.

Using a TFTP client application, you direct the upgrade by issuing write requests to send an upgrade file to the FastHub. After a successful transfer of the upgrade file, the FastHub resets and executes the new firmware.

Although the workstation-directed upgrade gives you extra flexibility, it can present a security issue. Because there is no file authentication involved, the existing FastHub firmware might be overwritten with outdated firmware. To prevent this, upgrade the firmware, and then set the MIB object mrUpgradeTFTPAccept to disabled; the FastHub ignores future workstation-directed upgrade requests.

## Out-Of-Band Upgrade

The XMODEM protocol is used to perform this upgrade. Refer to the *Catalyst FastHub 100+ Series Installation and Configuration Guide* for detailed information on using the out-of-band management interface to perform a firmware upgrade.

# FastHub MIB Implementation

This section provides brief descriptions of individual MIB implementations. Also included for each MIB group are the actions used to configure and manage the FastHub and the MIB objects associated with each action.

## MIB-II

The Internet Activities Board recommends that all TCP/IP implementations be network-manageable. Aspects of managing various components of the Internet are specified in RFC 1156, *Management Information Base for Network Management of TCP/IP-based Internets*. This MIB is referred to as the MIB-II.

The FastHub implements all groups in the MIB-II except the Exterior Gateway Protocol (EGP) group. MIB-II objects are used to control and monitor the management protocol operations of the NMM SNMP agent.

**Note** The MIB-II is not documented in this manual. See RFC 1213.

## Ethernet MIB

The standard Ethernet transmission MIB is used to supplement the MIB-II.

**Note** The Ethernet MIB is not documented in this manual. See RFC 1643.

# RS-232 MIB

**Note** EIA/TIA-232 was known as the recommended standard RS-232 before its acceptance as a standard by the Electronic Industries Association (EIA) and Telecommunications Industry Association (TIA). Because RS-232 appears in the names of supported MIB objects, this manual also uses RS-232.

The RS-232 MIB is used to configure the NMM RS-232 serial port (console port).

| Action | Associated MIB Objects |
|---|---|
| View/Configure RS-232 Port Characteristics | rs232Number<br>rs232PortIndex<br>rs232PortType<br>rs232PortInSigNumber<br>rs232PortOutSigNumber<br>rs232PortInSpeed<br>rs232PortOutSpeed |
| View RS-232 Port Input/Output Signals | rs232InSigPortIndex<br>rs232InSigName<br>rs232InSigState<br>rs232InSigChanges<br>rs232OutSigPortIndex<br>rs232OutSigName<br>rs232OutSigState<br>rs232OutSigChanges |
| View/Configure RS-232 Async Port Characteristics | rs232AsyncPortIndex<br>rs232AsyncPortBits<br>rs232AsyncPortStopBits<br>rs232AsyncPortParity<br>rs232AsyncPortAutobaud |
| View RS-232 Async Port Statistics | rs232AsyncPortParityErrs<br>rs232AsyncPortFramingErrs<br>rs232AsyncPortOverrunErrs |

## Repeater MIB

This is the standard Repeater MIB for managing IEEE 802.3 repeaters. The FastHub supports this MIB as it is defined in RFC 1516, *Definitions of Managed Objects for IEEE 802.3 Repeater Devices*.

| Action | Associated MIB Objects |
|---|---|
| View FastHub Operational Status | rptrGroupCapacity<br>rptrOperStatus<br>rptrHealthText<br>rptrTotalPartitionedPorts<br>rptrMonitorTransmitCollisions |
| Reset/Test FastHub | rptrReset<br>rptrNonDisruptTest |
| View/Configure FastHub Ports | rptrPortGroupIndex<br>rptrPortIndex<br>rptrPortAdminStatus<br>rptrPortAutoPartitionState<br>rptrPortOperStatus<br>rptrPortConnectorType<br>rptrPortLinkbeatStatus<br>rptrPortName |
| View/Configure FastHub Groups | rptrGroupIndex<br>rptrGroupDescr<br>rptrGroupObjectID<br>rptrGroupOperStatus<br>rptrGroupLastOperStatusChange<br>rptrGroupPortCapacity |
| View FastHub Group Statistics | rptrMonitorGroupIndex<br>rptrMonitorGroupTotalFrames<br>rptrMonitorGroupTotalOctets<br>rptrMonitorGroupTotalErrors |

| Action | Associated MIB Objects |
|---|---|
| View FastHub Port Statistics | rptrMonitorPortGroupIndex<br>rptrMonitorPortIndex<br>rptrMonitorPortReadableFrames<br>rptrMonitorPortReadableOctets<br>rptrMonitorPortFCSErrors<br>rptrMonitorPortAlignmentErrors<br>rptrMonitorPortFrameTooLongs<br>rptrMonitorPortShortEvents<br>rptrMonitorPortRunts<br>rptrMonitorPortCollisions<br>rptrMonitorPortLateEvents<br>rptrMonitorPortVeryLongEvents<br>rptrMonitorPortDataRateMismatches<br>rptrMonitorPortAutoPartitions<br>rptrMonitorPortTotalErrors<br>rptrMonitorPortIsolates<br>rptrMonitorPortSymbolErrors |
| View Address Tracking Information | rptrAddrTrackGroupIndex<br>rptrAddrTrackPortIndex<br>rptrAddrTrackLastSourceAddress<br>rptrAddrTrackSourceAddrChanges<br>rptrAddrTrackNewLastSrcAddress |

## Enterprise-Specific MIB

The FastHub implements extensions to the standard MIB-II in the form of the enterprise-specific MIB. These extensions are used to manage unique characteristics of the FastHub architecture.

| Action | Associated MIB Objects |
|---|---|
| View/Configure Hub Stack | mrStackUnit Capacity<br>mrStackNumberofUnitsPresent<br>mrStackSelectPrimarySupervisorUnit<br>mrStackUnitSupervisorIsPrimary |
| Clear Stack Statistics | mrStackClearStatistics |

| Action | Associated MIB Objects |
|---|---|
| View/Configure POST | mrStackPOSTSelect<br>mrStackUnitPOSTResult |
| Reset Hub Stack | mrStackReset<br>mrStackDefaultReset |
| View/Configure Supervisor Log | mrSupervisorClearLogTable<br>mrSupervisorLogIndex<br>mrSupervisorLogTime<br>mrSupervisorLogInfo |
| View/Configure FastHub (Unit) | mrStackUnitIndex<br>mrStackUnitPresent<br>mrStackUnitFirstGroupIndex<br>mrStackUnitLastGroupIndex<br>mrStackUnitSupervisorPresent<br>mrStackUnitSupervisorMajorVersion<br>mrStackUnitSupervisorMinorVersion<br>mrStackUnitSupervisorBootstrapMajorVersion<br>mrStackUnitSupervisorBootstrapMinorVersion<br>mrStackUnitPortVisualIndicatorSelect<br>mrStackUnitBasePortVisualIndicatorGreenMap<br>mrStackUnitBasePortVisualIndicatorAmberMap<br>mrStackUnitActivityVisualIndicator<br>mrStackUnitCollisionVisualIndicator |
| View/Configure 100BaseTX/16 Port Expansion Module | mrStackUnitExpansionModulePresent<br>mrStackUnitExpansionPortVisualIndicatorGreenMap<br>mrStackUnitExpansionPortVisualIndicatorAmberMap |
| View Unit Redundant Power Supply (RPS) Status | mrStackUnitRPSStatus<br>mrStackUnitRPSVisualIndicator |
| View/Configure Network Management Module (NMM) | mrNetMgmtIpAddress<br>mrNetMgmtIpSubnetMask<br>mrNetMgmtDefaultGateway<br>mrNetMgmtEnableAuthenTraps<br>mrNetMgmtEnableRIP |
| View/Configure Domain Name Servers | mrNetMgmtDomainServer1IpAddress<br>mrNetMgmtDomainServer2IpAddress<br>mrNetMgmtDefaultSearchDomain |

| Action | Associated MIB Objects |
|---|---|
| Configure the Management Console | mrNetMgmtConsoleInactTime<br>mrNetMgmtConsolePasswordThreshold<br>mrNetMgmtConsoleSilentTime |
| View/Configure Set Clients | mrNetMgmtSetClientIndex<br>mrNetMgmtSetClientName<br>mrNetMgmtSetClientStatus |
| View/Configure Trap Clients and Traps | mrNetMgmtTrapClientIndex<br>mrNetMgmtTrapClientName<br>mrNetMgmtTrapClientComm<br>mrNetMgmtTrapClientStatus<br>logonIntruder<br>hubStackDiagnostic<br>powerSupplyFailure<br>ipAddressChange |
| Configure a Modem (RS-232 port) | mrNetMgmtModemInitString<br>mrNetMgmtModemDialString<br>mrNetMgmtModemDialDelay<br>mrNetMgmtModemAutoAnswer |
| Upgrade FastHub Firmware | mrUpgradeFlashSize<br>mrUpgradeLastUpgradeTime<br>mrUpgradeLastUpgradeSource<br>mrUpgradeLastUpgradeStatus<br>mrUpgradeTFTPServerAddress<br>mrUpgradeTFTPLoadFilename<br>mrUpgradeTFTPInitiate<br>mrUpgradeTFTPAccept |

## Remote Monitoring MIB

The FastHub implements the first four object groups of the standard Remote Monitoring (RMON) MIB.

| Action | Associated MIB Objects |
|---|---|
| View/Configure Ethernet Statistics Group | etherStatsIndex<br>etherStatsDataSource<br>etherStatsDropEvents<br>etherStatsOctets<br>etherStatsPkts<br>etherStatsBroadcastPkts<br>etherStatsMulticastPkts<br>etherStatsCRCAlignErrors<br>etherStatsUndersizePkts<br>etherStatsOversizePkts<br>etherStatsFragments<br>etherStatsJabbers<br>etherStatsCollisions<br>etherStatsPkts64Octets<br>etherStatsPkts65to127Octets<br>etherStatsPkts128to255Octets<br>etherStatsPkts256to511Octets<br>etherStatsPkts512to1023Octets<br>etherStatsPkts1024to1518Octets<br>etherStatsOwner<br>etherStatsStatus |
| View/Configure History Control Group | historyControlIndex<br>historyControlDataSource<br>historyControlBucketsRequested<br>historyControlBucketsGranted<br>historyControlInterval<br>historyControlOwner<br>historyControlStatus |

| Action | Associated MIB Objects |
|---|---|
| View History Group Statistics | etherHistoryIndex<br>etherHistorySampleIndex<br>etherHistoryIntervalStart<br>etherHistoryDropEvents<br>etherHistoryOctets<br>etherHistoryPkts<br>etherHistoryBroadcastPkts<br>etherHistoryMulticastPkts<br>etherHistoryCRCAlignErrors<br>etherHistoryUndersizePkts<br>etherHistoryOversizePkts<br>etherHistoryFragments<br>etherHistoryJabbers<br>etherHistoryCollisions<br>etherHistoryUtilization |
| View/Configure Alarm Group | alarmIndex<br>alarmInterval<br>alarmVariable<br>alarmSampleType<br>alarmValue<br>alarmStartupAlarm<br>alarmRisingThreshold<br>alarmFallingThreshold<br>alarmRisingEventIndex<br>alarmFallingEventIndex<br>alarmOwner<br>alarmStatus |
| View/Configure Event Group | eventIndex<br>eventDescription<br>eventType<br>eventCommunity<br>eventLastTimeSent<br>eventOwner<br>eventStatus<br>logEventIndex<br>logIndex<br>logTime<br>logDescription |

| Action | Associated MIB Objects |
|---|---|
| View Remote Monitoring Traps | risingAlarm<br>fallingAlarm |

## Cisco Discovery Protocol MIB

The CDP MIB is a media- and protocol-independent device-discovery protocol that runs on all Cisco-manufactured equipment including routers, access servers, bridges, and switches. Using CDP, a device can advertise its existence to other devices and receive information about other devices on the same local-area network (LAN) or on the remote side of a wide-area network (WAN). CDP runs on all media that support Subnetwork Access Protocol (SNAP), including LANs, Frame Relay, and Asynchronous Transfer Mode (ATM) media.

| Action | Associated MIB Objects |
|---|---|
| View/Configure CDP Interfaces | cdpInterfaceIfIndex<br>cdpInterfaceEnable<br>cdpInterfaceMessageInterval<br>cdpInterfaceGroup<br>cdpInterfacePort |
| View CDP Cache | cdpCacheIfIndex<br>cdpCacheDeviceIndex<br>cdpCacheAddressType<br>cdpCacheAddress<br>cdpCacheVersion<br>cdpCacheDeviceId<br>cdpCacheDevicePort<br>cdpCachePlatform<br>cdpCacheCapabilities |