

INTERNET-DRAFT

[draft-anderson-docs-rf-mib-00.txt](#)

Expiration Date: October, 19, 1997

P. Anderson, CableLabs

W. Sawyer, Bay Networks

R. Woundy, Continental Cablevision

Data Over Cable Service (DOCS)  
Radio Frequency (RF) Interface Management Information Base (MIB)

Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as ``work in progress.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

This Internet-Draft outlines the Radio Frequency (RF) Interface Management Information Bases (MIBs) for high-speed data-over-cable systems developed by the MCNS Data Over Cable Services working group.

Two Simple Network Management Protocol (SNMP) MIBs are defined. The first is the MCNS Interface MIB and defines objects that enable management of the CATV MAC and PHY layer interfaces. The second is the MCNS Cable Modem (CM) MIB and defines objects that enable management of CMs and Cable Modem Termination Systems (CMTSs).

## Table of Contents

<a href="#">1. Introduction.....</a>	<a href="#">2</a>
<a href="#">2. CM and CMTS Management Requirements.....</a>	<a href="#">2</a>
<a href="#">2.1 Accounting Management.....</a>	<a href="#">2</a>
<a href="#">2.2 Configuration Management.....</a>	<a href="#">3</a>
<a href="#">2.2.1 Version Control.....</a>	<a href="#">3</a>
<a href="#">2.2.2 Software upgrades.....</a>	<a href="#">3</a>
<a href="#">2.2.3 System Initialization and Configuration.....</a>	<a href="#">3</a>
<a href="#">2.3 Fault Management.....</a>	<a href="#">4</a>
<a href="#">2.3.1 SNMP Usage.....</a>	<a href="#">4</a>
<a href="#">2.3.2 Event Logging.....</a>	<a href="#">4</a>
<a href="#">2.3.3 Trap and Syslog Throttling.....</a>	<a href="#">5</a>
<a href="#">2.3.4 Non-SNMP Fault Management protocols.....</a>	<a href="#">6</a>
<a href="#">2.4 Performance Management.....</a>	<a href="#">6</a>
<a href="#">2.5 Protocol Filters.....</a>	<a href="#">7</a>
<a href="#">2.6 Common Spectrum Management.....</a>	<a href="#">7</a>
<a href="#">3. Areas for Future Consideration.....</a>	<a href="#">8</a>
<a href="#">4. Management Information Base (MIB) .....</a>	<a href="#">8</a>
<a href="#">4.1 MIB Organization.....</a>	<a href="#">8</a>
<a href="#">4.2 Managed Objects from Existing Standards.....</a>	<a href="#">9</a>
<a href="#">4.2.1 The MIB-II 'system' Group.....</a>	<a href="#">9</a>
<a href="#">4.2.2 The MIB-II 'interface' Group.....</a>	<a href="#">9</a>
<a href="#">4.2.3 The MIB-II 'ip' Group.....</a>	<a href="#">10</a>
<a href="#">4.2.4 Other MIB-II Groups.....</a>	<a href="#">11</a>
<a href="#">4.2.5 The Bridge MIB.....</a>	<a href="#">11</a>
<a href="#">4.2.6 The Ethernet-Like MIB.....</a>	<a href="#">11</a>
<a href="#">4.2.7 The FDDI MIB.....</a>	<a href="#">11</a>
<a href="#">5. Concise MIB Definition.....</a>	<a href="#">11</a>
<a href="#">5.1 Root MIB for MCNS.....</a>	<a href="#">11</a>
<a href="#">5.2 Definition of Managed Objects for Data-over-Cable</a>	
Service Interfaces.....	<a href="#">12</a>
<a href="#">5.3 Definitions of Managed Objects for Cable Modems</a>	
and Cable Modem Termination Systems .....	<a href="#">50</a>
<a href="#">6. Protocol Definition for SYSLOG .....</a>	<a href="#">75</a>
<a href="#">7. References.....</a>	<a href="#">76</a>
<a href="#">8. Authors' Addresses.....</a>	<a href="#">77</a>

Anderson, Sawyer, Woundy

[Page 1]

## **1. Introduction**

This document outlines the Radio Frequency (RF) Interface Management Information Bases (MIBs) for high-speed data-over-cable systems developed by the MCNS Data Over Cable Services working group.

Two Simple Network Management Protocol (SNMP) MIBs are defined. The first is the MCNS Interface MIB and defines objects that enable management of the CATV MAC and PHY layer interfaces. The second is the MCNS Cable Modem (CM) MIB and defines objects that enable management of CMs and Cable Modem Termination Systems (CMTSs).

This specification is intended to enable prospective vendors of cable modems and other data-over-cable systems to address the operations support requirements in a uniform and consistent manner.

## **2. CM and CMTS Management Requirements**

This section describes the CM and CMTS management requirements. The MIBs compliant with these requirements are described in [Section 3](#) and formally defined in [Section 5](#).

### **2.1 Accounting Management**

Although many different types of billing scenarios exist for operators, the only scenarios which require use of CM and CMTS managed objects are those based on metered usage or reserved bandwidth.

Common practice by several Internet Service Providers (ISPs) allows usage-based billing based on peak rates. A DOCS provider can implement usage-based billing two ways: By polling the CMs or by polling the CMTS.

In the first method, a service provider can poll the ifInOctets and ifOutOctets counters from the MIB-II [[RFC-1213](#)] Interfaces group on each CM. This has the advantage of enabling both upstream and downstream traffic metering with the potential disadvantage of affecting network performance.

The second metered billing method involves monitoring the mcnsifCmtsServiceTotalDataSlots counter from the mcnsifCmtsServiceTable on each CMTS. This has the advantage of avoiding congestion on the RF network; however, it enables upstream traffic metering only.

In a typical ISP environment, a BSS polls the appropriate counters on each customer device once every 15 minutes throughout a monthly billing cycle. This data is converted into an average utilization rate for the sample period. Doing so permits the ISP to bill based on peak bandwidth by choosing the sample ranked at the 90-95th percentile. Note that the billing system may also include time-of-day rate variations.

The billing of reserved upstream MAC bandwidth is aided by information available from the mcnsifQosProfileTable for each CM. These MIB

variables report the upstream QoS characteristics, not just the nominal bandwidth, associated with each service ID and enable the service provider to bill for Grade of Service by verifying QoS.

## **2.2 Configuration Management**

### **2.2.1 Version Control**

The CM MUST (and the CMTS SHOULD) support software revision and operational parameter configuration interrogation. In particular, the fields of the sysObjectID Object Identifier (OID) of the CM MUST successively encode the vendor ID, the hardware platform, the hardware revision, the software/PROM major revision number, the software/PROM minor revision number, and (optionally) the software patch level. Each parameter MUST occupy exactly one field. The fields of the sysObjectID OID of the CMTS SHOULD use the same encoding.

### **2.2.2 Software upgrades**

The CM software upgrade process is documented in [\[MCNS7\]](#). The mechanism to upgrade software from an SNMP manager MUST be supported by CMs, and SHOULD be supported by CMTSs. From a network management station, the operator:

- sets mcnscmSwServer to the address of the TFTP server for software upgrades
- sets mcnscmSwFilename to the file pathname of the software upgrade image
- sets mcnscmSwAdminStatus to upgrade-from-mgt

One reason for the SNMP-initiated upgrade is to allow loading of a temporary software image (e.g., special diagnostic software) that differs from the software normally used on that modem without changing the provisioning database.

Note that software upgrades should not be accepted blindly by the cable modem. The cable modem may refuse an upgrade if:

- The download is incomplete.
- The file contents are incomplete or damaged.
- The software is not intended for that hardware device (may include the case of a feature set that has not been purchased for this device).

### **2.2.3 System Initialization and Configuration**

Most system configuration of CMs is performed through a combination of CATV MAC, DHCP, and TFTP exchanges. These exchanges are defined in detail in the Radio Frequency Interfaces Specification [\[MCNS7\]](#). In

particular, to enable event logging through SYSLOG, the DHCP server sets the log server option [[RFC-1533](#)] to the address of the SYSLOG server.

## **2.3 Fault Management**

### **2.3.1 SNMP Usage**

In the DOCS environment, the goals of fault management are the remote detection, diagnosis, and correction of network problems. Therefore, the CM MUST support SNMP management traffic across both the Ethernet and CATV MAC interfaces. Access to these interfaces may be restricted to support policy goals (see the `mcnscmNmAccessTable`); i.e., the service provider can choose to disable this at anytime.

CM installation personnel can use SNMP queries from a station on the Ethernet to perform on-site CM and CATV MAC diagnostics and fault classification (note that this may require temporary provisioning of the CM from an Ethernet DHCP server). Further, future customer applications using SNMP queries can diagnose simple post-installation problems, avoiding visits from service personnel and minimizing help desk telephone queries.

Standard MIB-II support MUST be implemented to instrument interface status, packet corruption, protocol errors, network performance, etc. The transmission MIB for Ethernet-like objects [[RFC-1643](#)] MUST be implemented on each CM and CMTS Ethernet and Fast Ethernet port. The `ifXTable` [[RFC-1573](#)] SHOULD be implemented to provide discrimination between broadcast and multicast traffic.

The CM and CMTS MUST support managed objects for fault management of the PHY and MAC layers. The MIB includes variables to track PHY state such as codeword collisions and corruption, signal-to-noise ratios, transmit and receive power levels, propagation delays, micro-reflections, in channel response, and Sync loss. The MIB also includes variables to track MAC state such as collisions and excessive retries for requests, immediate data transmits, and initial ranging requests.

For fault management at all layers, the CM/CMTS MUST generate replies to SNMP queries (subject to policy filters) for counters and status, MUST send SNMP traps to one or more trap NMSs (subject to policy), and MUST send event logging to a SYSLOG server (if a log server is defined). The `ifTestTable` [[RFC-1573](#)] SHOULD be implemented for any diagnostic test procedures that can be remotely initiated.

### **2.3.2 Event Logging**

Event logging and history provide vendors an opportunity for product differentiation. The ability to report useful logs may depend on semi-graceful failure modes and on the ability to record such in nonvolatile storage.





Events SHOULD be reported via log entries in a MIB, the SYSLOG facility (as documented in [Section 6](#)), and SNMP traps. Reporting of events SHOULD be fully configurable by priority class. At a minimum, it MUST be possible to disable SNMP Trap and SYSLOG transmission.

A local event log that is available via SNMP queries SHOULD be implemented to track events that cannot be reported at the time that they occur. This log SHOULD support a minimum of ten event log entries, and SHOULD persist across device re-boots.

The definition and coding of events is vendor-specific. In deference to the network operator who must troubleshoot multi-vendor networks, the circumstances and meaning of each event are reported as human-readable text. Vendors SHOULD provide time-of-day clocks in CMs to provide useful timestamping of events. Similarly, event logs SHOULD be persistent across device re-boots. The depth of the event log is vendor dependent, with oldest entries discarded as needed.

For each vendor-specific event that is reportable via TRAP, the vendor must create an enterprise-specific trap definition. Trap definitions MUST include mcnscmEvText and should be defined as:

```
trapName          TRAP-TYPE
                  ENTERPRISE  myEnterprise
                  VARIABLES    { mcnscmEvText, other useful objects }
                  DESCRIPTION
                      "trap description"
                      ::= value of mcnscmEvId
```

The event framework described in this section MUST be implemented in CMs and SHOULD be implemented in CMTSS.

### **[2.3.3](#) Trap and Syslog Throttling**

The CM and CMTS MUST provide support for trap and syslog message throttling as described below. The network operator can employ message rate throttling or trap limiting by manipulating the appropriate MIB variables.

#### **[2.3.3.1](#) Rate Throttling**

Network operators may employ either of two rate control methods. In the first method, the device ceases to send traps and SYSLOG messages when the rate exceeds the specified maximum message rate. It resumes sending traps only if reactivated by a network management station request.

In the second method, the device resumes sending traps when the rate falls below the specified maximum message rate.



The network operator configures the specified maximum message rate by setting the measurement interval (in seconds), and the maximum number of trap and SYSLOG messages (excluding duplicates) to be transmitted within the measurement interval. The operator can query the operational throttling state (to determine whether traps are enabled or blocked by throttling) of the device, as well as query and set the administrative throttling state (to manage the rate control method) of the device.

#### **2.3.3.2 Trap Limiting**

Network operators may wish to limit the number of traps sent by a device over a specified time period. The device ceases to send traps and SYSLOG messages when the number of traps exceeds the specified threshold. It resumes sending traps only when the measurement interval has passed.

The network operator defines the maximum number of traps he is willing to handle and sets the measurement interval to a large number (in hundredths of a second). For this case, the administrative throttling state is set to stop at threshold which is the maximum number of traps.

See "Techniques for Managing Asynchronously Generated Alerts" [[RFC-1224](#)] for further information.

#### **2.3.4 Non-SNMP Fault Management protocols**

The OSS can use a variety of tools and techniques to examine faults at multiple layers. For the IP layer, useful non-SNMP based tools include ping (ICMP Echo and Echo Reply), traceroute (UDP and various ICMP Destination Unreachable flavors). Pings to a CM from its Ethernet side MUST be supported to enable local connectivity testing from a customer's PC to the modem. The CM and CMTS MUST support IP end-station generation of ICMP error messages and processing of all ICMP messages.

### **2.4 Performance Management**

At the CATV MAC layer, performance management focuses on the monitoring of the effectiveness of cable plant segmentation and rates of upstream traffic and collisions. Instrumentation is provided in the form of the standard interfaces statistics, as well as the `mcnsifCmtsServiceTable` and `mcnsifCmServiceTable`.

It is not anticipated that the CMTS upstream bandwidth allocation function will require active network management intervention and tuning. Nevertheless, management objects are provided in case tuning or direct control is necessary. The three key upstream contention intervals are the request interval, the immediate data interval, and the initial ranging maintenance interval. If the upstream collision rate of requests and immediate data is high relative to the upstream traffic bandwidth, then the network management system (NMS) might increase the size of the

request and immediate data intervals, respectively. The NMS might increase the size of the initial ranging maintenance interval when the

upstream collision rate of initial ranging messages is relatively high, such as at the conclusion of a wide-spread regional power outage. The NMS might also decrease the size of these contention intervals under low collision rate conditions, since these intervals occupy bandwidth that may be otherwise used for upstream transmission bandwidth. As a last resort, the NMS might change the guaranteed upstream bandwidth for one or more service IDs, to relieve upstream traffic congestion for key subscribers. The CM MUST implement MIB counters that report the number of contention interval collisions (measured as the number of contention interval retries) per service ID, and the CMTS MUST implement read-write MIB objects that control the size of the contention intervals for each upstream channel. The CMTS SHOULD implement a read-write MIB object that controls the guaranteed upstream bandwidth for each service ID.

At the LLC layer, the performance management focus is on bridge traffic management. The CM and CMTS (if the CMTS implements transparent bridging) MUST implement the Bridge MIB [[RFC-1493](#)], including the dot1dBase and dot1dTp groups. The CM and CMTS MUST implement a managed object that controls whether the 802.1d spanning tree protocol (STP) is run and topology update messages are generated; STP is unnecessary in hierarchical, loop-free topologies. If the STP is enabled for the CM/CMTS, then the CM/CMTS MUST implement the dot1dStp group. These MIB groups' objects allow the NMS to detect when bridge forwarding tables are full, and enable the NMS to modify aging timers.

A final performance concern is the ability to diagnose unidirectional loss. Both the CM and CMTS MUST implement the MIB-II [[RFC-1213](#)] Interfaces group. When there exists more than one upstream or downstream channel, the CM/CMTS MUST implement an instance of IfEntry for each channel. The ifStack MIB [[RFC-1573](#)] MUST be used to define the relationships among the CATV MAC interfaces and their channels.

## **[2.5 Protocol Filters](#)**

The CM MUST implement LLC and IP protocol filters. The LLC protocol filter entries can be used to limit CM forwarding to a restricted set of network-layer protocols (such as IP, IPX, NetBIOS, and Appletalk). The IP protocol filter entries can be used to restrict upstream or downstream traffic based on source and destination IP addresses, transport-layer protocols (such as TCP, UDP, and ICMP), and source and destination TCP/UDP port numbers. The CM SHOULD support a minimum of ten LLC protocol filter entries, and ten IP protocol filter entries.

## **[2.6 Common Spectrum Management](#)**

The CMTS SHOULD implement the HFC RF Spectrum Management MIB [[CSMIMIB](#)]. The definition of this MIB is likely to evolve, and vendors should anticipate changes in this area.



### **3. Areas for Future Consideration**

This section outlines some areas for future consideration within this specification.

- As ATM standards develop, this specification will include appropriate standards references.
- Enterprise-specific traps will be defined in the future as dictated by field experience.
- Multicast service provisioning within the cable modem will be clearly defined.
- To support the billing of reserved downstream MAC bandwidth, the CMTS should implement the evolving RSVP/Integrated Services MIB(s). Because of the variety of output queuing mechanisms, comments are solicited for the management mechanisms to support this.
- Comments are solicited as to required and recommended diagnostic test procedures.

### **4. Management Information Base (MIB)**

This section defines the minimum set of managed objects required to support the CM and CMTS management requirements identified in the previous section. Vendors may augment this MIB with objects from other standard or vendor-specific MIBs where appropriate.

#### **4.1 MIB Organization**

There are three parts of the MIB needed for CMs and CMTSs. The first is a set of objects drawn from standard SNMP MIBs that bear on this class of devices. It is not the intention of this specification to duplicate existing specifications. These are available as RFCs from the IETF and are widely available.

The second part is a set of objects for the CATV interfaces of the CM and CMTS. This MIB provides the objects needed to configure, operate, and monitor the physical CATV interfaces. This specification is derived from the MCNS MAC specification [[MCNS7](#)]. These objects are defined in [Section 5.2](#) as the mcnsif MIB.

The third part is a set of objects for management of CM and CMTS devices. These provide system-level functionality that is specific to the business and operational environments of cable data systems. These objects are defined in [Section 5.3](#) as the mcnscm MIB.

The latter two parts are presented here as separate MIBs. The mcnsif and mcnscm MIBs are formally defined in [Section 5](#).





4.2 Managed Objects from Existing Standards

4.2.1 The MIB-II 'system' Group

The System Group from MIB-II [RFC-1213] MUST be implemented. See Section 2.2.1 for sysObjectID requirements.

4.2.2 The MIB-II 'interface' Group

The MIB-II interface group provides essential information about both MAC interfaces and individual channels. The interface extensions (ifXTable) [RFC-1573] SHOULD be supported.

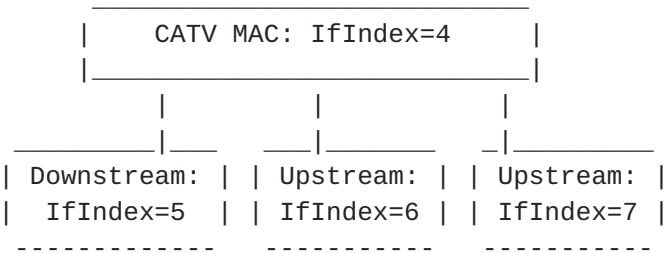
4.2.2.1 Interface Organization and Numbering

An instance of ifEntry exists for each CATV MAC interface, as well as each LAN interface.

If a MAC interface consists of more than one upstream and downstream channel, then a separate instance of ifEntry MUST also exist for each channel.

If table entries exist separately for upstream and downstream channels, then the ifStack group ([RFC-1573]) must be implemented to identify relationships among sub-interfaces. Note that the CATV MAC interface must exist, even if it is further broken out into sub-interfaces.

The example in Figure 4-1 illustrates a MAC interface with one downstream and two upstream channels:



Implementation of ifStackTable for this example.

IfStackHigherLayer	ifStackLowerLayer
4	5
4	6
4	7

Figure 4-1: Interface Numbering Example



At the CMTS, interface numbering is at the discretion of the vendor, and should correspond to the physical arrangement of connections. If table entries exist separately for upstream and downstream channels, then the ifStack group ([\[RFC-1573\]](#)) must be implemented to identify relationships among sub-interfaces. Note that the CATV MAC interface(s) must exist, even if further broken out into sub-interfaces.

At the CM, interfaces SHOULD be numbered as:

- Ethernet: 1
- CATV MAC: 2
- all others (individual channels if present, telephony return if present): n+2

Note that network management platforms MUST NOT rely on this numbering scheme, and should use ifType and the ifStack table to identify interfaces.

#### **[4.2.2.2](#) Specific Interface Attributes**

The ifAdminStatus object provides administrative control over both MAC interfaces and individual channels.

For CATV MAC interfaces, ifSpeed is defined as the bit rate of the highest-speed channel which is attached to this interface.

The ifSpecific object must be set to { mcnsif } for CATV MAC interfaces. For upstream channels, it is set to { mcnsifUpstreamTable }. For downstream channels, it is set to { mcnsifDownstreamTable }. Note that this object is deprecated in reference [\[RFC-1573\]](#).

The ifType object needs enumerated values to be assigned from the Internet Assigned Numbers Authority. Values are needed for:

- CATV CMTS MAC interface
- CATV CM MAC interface
- CATV upstream channel
- CATV downstream channel

For each Ethernet or Fast Ethernet interface present on a CM or CMTS, there must exist a corresponding ifEntry with ifType ethernet-csmacd(6) [\[RFC1213\]](#).

#### **[4.2.3](#) The MIB-II 'ip' Group**

The IP group is implemented. It does not apply to IP packets forwarded by the device as a link-layer bridge. For the CM, it applies only to the device as an IP host. At the CMTS, it applies to the device as an IP host, and as a router if IP routing is implemented.



#### [4.2.4](#) Other MIB-II Groups

All devices MUST implement all MIB-II groups which apply to the functionality of the device. At a minimum, this includes the ICMP, UDP, and SNMP groups, all of which apply only to the device's role as an IP host.

#### [4.2.5](#) The Bridge MIB

In both the CM and the CMTS (if the CMTS implements transparent bridging), the Bridge MIB [[RFC-1493](#)] MUST be implemented to represent the bridging process.

In the CMTS that implements transparent bridging, the Bridge MIB SHOULD be used to represent information about the MAC Forwarder state.

#### [4.2.6](#) The Ethernet-Like MIB

The Ethernet-like MIB ([[RFC-1643](#)]) must be implemented if Ethernet or Fast Ethernet interfaces are present.

#### [4.2.7](#) The FDDI MIB

The FDDI MIB ([[RFC-1512](#)]) must be implemented if Fiber Distributed Data Interfaces are present.

### [5.](#) Concise MIB Definition

This section contains formal definitions of the MCNS Interface MIB and the MCNS CM MIB. Both are presented in the SNMP Version 2 Concise MIB Definition format. An essentially empty MCNS MIB provides a root point for the two MIBs.

#### [5.1](#) Root MIB for MCNS

This MIB exists solely to provide an anchor in the mib tree for the two MIBs which follow.

```
MCNS-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
MODULE-IDENTITY,
enterprises
    FROM SNMPv2-SMI;
```

```
mcns MODULE-IDENTITY
LAST-UPDATED "9703241130Z"
ORGANIZATION "Multimedia Cable Network System"
```



CONTACT-INFO " Pam Anderson  
Postal: CableLabs  
400 Centennial Parkway  
Louisville, CO 80027-1266  
Tel: +1 303 661 9100  
Fax: +1 303 661 9199  
E-mail: p.anderson@cablelabs.com"

**DESCRIPTION**

"This is the MIB Module for MCNS-compliant cable modems and cable-modem termination systems."

::= { enterprises 2141 }

END

## **5.2 Definition of Managed Objects for Data-over-Cable Service Interfaces**

The following groups are provided for management of the CATV MAC layer interfaces:

-mcnsifCmMacTable (and mcnsifCmtsMacTable) - each entry extends the definition of the CM (and CMTS) MAC interface beyond that included in ifEntry. The CMTS table provides controls for bandwidth allocation and other aspects of network control.

-mcnsifDownstreamChannelTable - each entry describes the characteristics of a single downstream channel.

-mcnsifUpstreamChannelTable - each entry describes the characteristics of a single upstream channel.

-mcnsifPreambleTable - provides reference preamble strings for use with mcnsifModulationTable.

-mcnsifModulationTable - each entry describes a modulation profile associated with one or more upstream channels.

-mcnsifCmServiceTable (and mcnsifCmtsServiceTable) - describes each service queue. Each CM will maintain one or more service queues. Each CMTS will assign and maintain all of the service queues within a MAC domain.

-mcnsifQosProfileTable - defines quality of service characteristics to be used by the mcnsifServiceTable.

-mcnsifCmtsTallyTable - provides MAC-layer protocol tallies for the CMTS.

-mcnsifCmTallyTable - provides MAC-layer protocol tallies for the CM.





-mcnsifSignalQualityTable - provides information about physical-layer signal quality.

CableModemInterface-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY, OBJECT-TYPE,  
Integer32, Counter32, TimeTicks, IpAddress

FROM SNMPv2-SMI

TEXTUAL-CONVENTION, MacAddress, RowStatus  
FROM SNMPv2-TC

OBJECT-GROUP, MODULE-COMPLIANCE  
FROM SNMPv2-CONF

ifIndex

FROM [RFC1213](#)-MIB

mcns

FROM MCNS-MIB

;

mcnsifMIB MODULE-IDENTITY

LAST-UPDATED "9703241130Z"

ORGANIZATION "Multimedia Cable Network System"

CONTACT-INFO " Pam Anderson

Postal: CableLabs

400 Centennial Parkway

Louisville, CO 80027-1266

Tel: +1 303 661 9100

Fax: +1 303 661 9199

E-mail: p.anderson@cablelabs.com"

DESCRIPTION

"This is the MIB Module for MCNS-compliant MAC interfaces at cable modems and cable-modem termination systems."

::= { mcns 1 }

-- Textual Conventions

TenthdBmV ::= TEXTUAL-CONVENTION

DISPLAY-HINT "d-1"

STATUS current

DESCRIPTION

"This data type represents power levels that are normally expressed in dBmV. Units are in tenths of a dBmV; for example, **5.1 dBmV will be represented as 51.**"

SYNTAX INTEGER

Milliseconds ::= TEXTUAL-CONVENTION

DISPLAY-HINT "d"

STATUS        current

Anderson, Sawyer, Woundy

[Page 13]

## DESCRIPTION

"This data type represents time intervals in milliseconds."

SYNTAX INTEGER

mcnsifMIBObjects OBJECT IDENTIFIER ::= { mcnsifMIB 1 }

--

-- The CM MAC Table

--

mcnsifCmMacTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifCmMacEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes the attributes of each CM MAC interface, extending the information available from ifEntry. Mandatory for all Cable Modems."

::= { mcnsifMIBObjects 1 }

mcnsifCmMacEntry OBJECT-TYPE

SYNTAX McnsifCmMacEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry containing objects describing attributes of each MAC entry, extending the information in ifEntry."

INDEX { ifIndex }

::= { mcnsifCmMacTable 1 }

McnsifCmMacEntry ::= SEQUENCE {

mcnsifCmtsAddress MacAddress,

mcnsifCmCapabilities INTEGER,

mcnsifCmResets Counter32,

mcnsifCmRangingRespTimeout TimeTicks

}

mcnsifCmtsAddress OBJECT-TYPE

SYNTAX MacAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Identifies the CMTS that is believed to control this MAC domain. At the CM, this will be the source address from SYNC, MAP, and other MAC-layer messages. If the CMTS is unknown, returns 00-00-00-00-00-00."

::= { mcnsifCmMacEntry 1 }



mcnsifCmCapabilities OBJECT-TYPE

SYNTAX INTEGER {  
    completeFramesOnly(1),  
    atmCells(2),  
    concatenation(3),  
    concatenationAndAtm(4)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Identifies the capabilities of the MAC implementation at this interface."

::= { mcnsifCmMacEntry 2 }

mcnsifCmResets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A count of the number of times that the CM reset or initialized this interface."

::= { mcnsifCmMacEntry 3 }

mcnsifCmRangingRespTimeout OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Waiting time for a Ranging Response packet."

REFERENCE

"Data over Cable Radio Frequency Interface specification,  
[Section 7](#), timer T3"

DEFVAL { 20 }

::= { mcnsifCmMacEntry 4 }

--

-- The CMTS MAC Table

--

mcnsifCmtsMacTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifCmtsMacEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes the attributes of each CMTS MAC interface, extending the information available from ifEntry. Mandatory for all CMTS devices."

```
::= { mcnsifMIBObjects 2 }
```

mcnsifCmtsMacEntry OBJECT-TYPE

SYNTAX McnsifCmtsMacEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry containing objects describing attributes of each MAC entry, extending the information in ifEntry."

INDEX { ifIndex }

::= { mcnsifCmtsMacTable 1 }

```
McnsifCmtsMacEntry ::= SEQUENCE {
    mcnsifCmtsCapabilities  INTEGER,
    mcnsifSyncInterval      Milliseconds,
    mcnsifUcdInterval       Milliseconds,
    mcnsifMinMapSize        INTEGER,
    mcnsifMaxMapSize        INTEGER,
    mcnsifMaxFutureMap      INTEGER,
    mcnsifMaxServiceIds     INTEGER,
    mcnsifRqPerMap          Integer32,
    mcnsifDataPerMap        Integer32,
    mcnsifInsertionInterval TimeTicks,
    mcnsifInvitedRangingRetries Integer32
}
```

mcnsifCmtsCapabilities OBJECT-TYPE

SYNTAX INTEGER {  
 completeFramesOnly(1),  
 atmCells(2),  
 concatenation(3),  
 concatenationAndAtm(4)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Identifies the capabilities of the CMTS MAC implementation at this interface."

::= { mcnsifCmtsMacEntry 1 }

mcnsifSyncInterval OBJECT-TYPE

SYNTAX Milliseconds

MAX-ACCESS read-write -- see security note in section 3 of [\[MCNS11\]](#)

STATUS current

DESCRIPTION

"The interval between CMTS transmission of successive SYNC messages at this interface."

REFERENCE

" Data Over Cable Radio Frequency Interface Specification, [\[MCNS7\]](#), Sync Interval."



```
::= { mcsifCmtsMacEntry 2 }
```

mcnsifUcdInterval            OBJECT-TYPE  
SYNTAX    Milliseconds  
MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"The interval between CMTS transmission of successive Upstream  
Channel Descriptor messages for each upstream channel at this  
interface."  
REFERENCE  
"Data Over Cable Radio Frequency Interface Specification,  
[\[MCNS7\]](#), UCD Interval."  
::= { mcnsifCmtsMacEntry 3 }

mcnsifMinMapSize            OBJECT-TYPE  
SYNTAX    INTEGER (1..4096)  
MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"The minimum allocation map size, in terms of the number of  
mini-slots described by the MAP."  
::= { mcnsifCmtsMacEntry 4 }

mcnsifMaxMapSize            OBJECT-TYPE  
SYNTAX    INTEGER (1..4096)  
MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"The maximum allocation map size, in terms of the number of  
mini-slots described by the MAP."  
::= { mcnsifCmtsMacEntry 5 }

mcnsifMaxFutureMap          OBJECT-TYPE  
SYNTAX    INTEGER (1..4096)  
MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"The maximum number of mapped mini-slots at any given time."  
REFERENCE  
"Data Over Cable Radio Frequency Interface Specification,  
[\[MCNS7\]](#), Max Map Pending."  
::= { mcnsifCmtsMacEntry 6 }

mcnsifMaxServiceIds        OBJECT-TYPE  
SYNTAX    INTEGER (0..8191)  
MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"The maximum number of (unicast) service IDs that may be

simultaneously active. Requests for service beyond this limit  
should cause the CMTS to generate an event."  
::= { mcnsifCmtsMacEntry 7 }

mcnsifRqPerMap OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-write -- see security note in section 3 of [MCNS11]  
STATUS current  
DESCRIPTION  
"The number of request contention mini-slots to assign per  
allocation map. Zero indicates that a vendor-specific algorithm  
is used instead of a fixed number."  
::= { mcnsifCmtsMacEntry 8 }

mcnsifDataPerMap OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-write -- see security note in section 3 of [MCNS11]  
STATUS current  
DESCRIPTION  
"The number of data/request contention mini-slots to assign per  
allocation map. Zero indicates that a vendor-specific algorithm  
is used instead of a fixed number."  
::= { mcnsifCmtsMacEntry 9 }

mcnsifInsertionInterval OBJECT-TYPE  
SYNTAX TimeTicks  
MAX-ACCESS read-write -- see security note in section 3 of [MCNS11]  
STATUS current  
DESCRIPTION  
"The number of mini-slots to elapse between each grant of a  
broadcast station maintenance grant. Broadcast station  
maintenance grants are used to allow new cable modems to join  
the network. Zero indicates that a vendor-specific algorithm is  
used instead of a fixed number."  
REFERENCE  
"Data Over Cable Radio Frequency Interface Specification,  
[MCNS7], Ranging Interval."  
::= { mcnsifCmtsMacEntry 10 }

mcnsifInvitedRangingRetries OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-write  
STATUS current  
DESCRIPTION  
"The maximum number of retries to make on invitations for  
ranging requests."  
::= { mcnsifCmtsMacEntry 11 }

--  
-- The following table is implemented on both the CM and the CMTS  
-- On the CM, only the attached downstream channel is defined.  
--



mcnsifDownstreamChannelTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifDownstreamChannelEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes the attributes of each attached downstream channel (frequency band). Mandatory for all CM and CMTS devices."

::= { mcnsifMIBObjects 3 }

mcnsifDownstreamChannelEntry OBJECT-TYPE

SYNTAX McnsifDownstreamChannelEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"At the CMTS, describes the attributes of each downstream channel (frequency band). At the CM, describes the attached downstream channel."

INDEX { ifIndex }

::= { mcnsifDownstreamChannelTable 1 }

McnsifDownstreamChannelEntry ::= SEQUENCE {

mcnsifDownChannelId Integer32,

mcnsifDownChannelFrequency INTEGER,

mcnsifDownChannelModulation INTEGER,

mcnsifDownChannelInterleave INTEGER,

mcnsifDownChannelTxPower TenthdBmV

}

mcnsifDownChannelId OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The CMTS identification of the downstream channel within this particular MAC interface."

::= { mcnsifDownstreamChannelEntry 1 }

mcnsifDownChannelFrequency OBJECT-TYPE

SYNTAX INTEGER (910000000..860000000)

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The center of the frequency band associated with this channel, in Hertz."

::= { mcnsifDownstreamChannelEntry 2 }



**mcnsifDownChannelModulation OBJECT-TYPE**

```
SYNTAX  INTEGER {
            qam64(1),
            qam256(2)
        }
MAX-ACCESS      read-write          -- see security note in section 3 of [MCNS11]
STATUS          current
DESCRIPTION
    "The modulation type associated with this downstream channel."
 ::= { mcnsifDownstreamChannelEntry 3 }
```

**mcnsifDownChannelInterleave OBJECT-TYPE**

```
SYNTAX  INTEGER {
            otherUnknown(1),
            taps8Increment16(2), -- protection 5.9/4.1 usec, latency
            .22/.15 msec
            taps16Increment8(3), -- protection 12/8.2 usec, latency
            .48/.33 msec
            taps32Increment4(4), -- protection 24/16 usec, latency
            .98/.68 msec
            taps64Increment2(5), -- protection 47/33 usec, latency 2/1.4
            msec
            taps128Increment1(6) -- protection 95/66 usec, latency 4/2.8
            msec
        }
MAX-ACCESS      read-write          -- see security note in section 3 of [MCNS11]
STATUS          current
DESCRIPTION
    "The FEC interleaving used for this downstream channel."
 ::= { mcnsifDownstreamChannelEntry 4 }
```

**mcnsifDownChannelTxPower OBJECT-TYPE**

```
SYNTAX  TenthdBmV
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "At the CMTS, the operational transmit power. Set to zero by the
    CM."
 ::= { mcnsifDownstreamChannelEntry 5 }
```

--

-- The following table is implemented on both the CM and the CMTS. For  
-- the CM, only the channel to which it is attached appears in the  
-- table.

--

**mcnsifUpstreamChannelTable OBJECT-TYPE**

```
SYNTAX          SEQUENCE OF McnsifUpstreamChannelEntry
```



MAX-ACCESS not-accessible  
STATUS current

Anderson, Sawyer, Woundy

[Page 20]

## DESCRIPTION

"Describes the attributes of each attached upstream channel (frequency band). Mandatory for all CM and CMTS devices."

::= { mcnsifMIBObjects 4 }

mcnsifUpstreamChannelEntry OBJECT-TYPE

SYNTAX McnsifUpstreamChannelEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"At the CMTS, describes the attributes of each upstream channel (frequency band). At the CM, describes the attached upstream channel. At the CM, all table entries are read-only."

INDEX { ifIndex }

::= { mcnsifUpstreamChannelTable 1 }

```
McnsifUpstreamChannelEntry ::= SEQUENCE {  
    mcnsifUpChannelId      Integer32,  
    mcnsifUpChannelFrequency      INTEGER,  
    mcnsifUpChannelSymbolRate      INTEGER,  
    mcnsifUpChannelModulationProfile      Integer32,  
    mcnsifUpChannelSlotSize Integer32,  
    mcnsifUpChannelTxPower  TenthdBmV,  
    mcnsifUpChannelDownChannelTiming      Integer32,  
    mcnsifUpChannelTxTimingOffset Integer32,  
    mcnsifUpChannelRangingBackoffStart      Integer32,  
    mcnsifUpChannelRangingBackoffEnd      Integer32,  
    mcnsifUpChannelTxBackoffStart Integer32,  
    mcnsifUpChannelTxBackoffEnd Integer32  
}
```

mcnsifUpChannelId OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The CMTS identification of the upstream channel."

::= { mcnsifUpstreamChannelEntry 1 }

mcnsifUpChannelFrequency OBJECT-TYPE

SYNTAX INTEGER (5000000..42000000)

MAX-ACCESS read-write -- see security note in section 3 of [\[MCNS11\]](#)

STATUS current

## DESCRIPTION

"The center of the frequency band associated with this channel, in Hertz."

::= { mcnsifUpstreamChannelEntry 2 }



mcnsifUpChannelSymbolRate OBJECT-TYPE  
SYNTAX INTEGER (160000..2560000)  
MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The symbol rate of the upstream channel (determines the channel width)."  
::= { mcnsifUpstreamChannelEntry 3 }

mcnsifUpChannelModulationProfile OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"An entry identical to the mcnsifModIndex in the mcnsifModulationTable that describes this channel. This channel is further instantiated there by a grouping of interval usage codes which together fully describe the channel modulation."  
::= { mcnsifUpstreamChannelEntry 4 }

mcnsifUpChannelSlotSize OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The number of 6.25 microsecond ticks in each upstream mini-slot. This is read-only at the CM."  
::= { mcnsifUpstreamChannelEntry 5 }

mcnsifUpChannelTxPower OBJECT-TYPE  
SYNTAX TenthdBmV  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"At the CM, the operational transmit power. At the CMTS, always zero."  
::= { mcnsifUpstreamChannelEntry 6 }

mcnsifUpChannelDownChannelTiming OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The ID of the downstream channel providing timing (SYNC and UCD messages) for this upstream channel, from the point of view of the particular CM. Always zero for the CMTS."  
::= { mcnsifUpstreamChannelEntry 7 }



**mcnsifUpChannelTxTimingOffset OBJECT-TYPE**

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

**DESCRIPTION**

"A measure of the current round trip time at the CM, or the maximum round trip time seen by the CMTS. Used for timing of CM upstream transmissions to ensure synchronized arrivals at the CMTS. Units are in terms of (6.25 microseconds/64)."

::= { mcnsifUpstreamChannelEntry 8 }

**mcnsifUpChannelRangingBackoffStart OBJECT-TYPE**

SYNTAX Integer32

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

**DESCRIPTION**

"The initial random backoff window to use when retrying Ranging Requests. Expressed as a power of 2. Any value other than 0..15 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used."

::= { mcnsifUpstreamChannelEntry 9 }

**mcnsifUpChannelRangingBackoffEnd OBJECT-TYPE**

SYNTAX Integer32

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

**DESCRIPTION**

"The final random backoff window to use when retrying Ranging Requests. Expressed as a power of 2. Any value other than 0..15 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used."

::= { mcnsifUpstreamChannelEntry 10 }

**mcnsifUpChannelTxBackoffStart OBJECT-TYPE**

SYNTAX Integer32

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

**DESCRIPTION**

"The initial random backoff window to use when retrying transmissions. Expressed as a power of 2. Any value other than 0..15 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used."

::= { mcnsifUpstreamChannelEntry 11 }

**mcnsifUpChannelTxBackoffEnd OBJECT-TYPE**

SYNTAX Integer32

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current



## DESCRIPTION

"The final random backoff window to use when retrying transmissions. Expressed as a power of 2. Any value other than 0..15 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used."

::= { mcnsifUpstreamChannelEntry 12 }

--

-- The preamble table is used with the modulation profile table which follows. It provides a set of  
-- preamble strings that can then be referenced by any of the modulation profiles. The preamble strings  
-- transmitted in the MAC-layer Upstream Channel Descriptor message are drawn from this table.

--

mcnsifPreambleTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifPreambleEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Provides a set of baseline preamble strings to be used with various modulation profiles."

::= { mcnsifMIBObjects 5 }

mcnsifPreambleEntry OBJECT-TYPE

SYNTAX McnsifPreambleEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes a baseline preamble string for one or more upstream channels."

INDEX { mcnsifPreambleIndex }

::= { mcnsifPreambleTable 1 }

McnsifPreambleEntry ::= SEQUENCE {

    mcnsifPreambleIndex Integer32,

    mcnsifPreambleValue OCTET STRING

}

mcnsifPreambleIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Indexes the baseline preamble strings. Must match mcnsifModIndex."

::= { mcnsifPreambleEntry 1 }





mcnsifPreambleValue OBJECT-TYPE  
SYNTAX OCTET STRING (SIZE (0..128))  
MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"A string of up to 1024 bits that defines the baseline preamble pattern to prepend on this channel/IUC grouping when transmitting. The bit pattern is left justified in the string and unused low order bits of the last octet are padded with zeros. Various modulation profiles will choose substrings of this value. This object is read-only at the CM."  
::= { mcnsifPreambleEntry 2 }

--  
-- The following table provides upstream channel modulation profiles. Entries in this table can be  
-- re-used by one or more upstream channels. An upstream channel will have a modulation profile  
-- for each value of mcnsifModIntervalUsageCode.  
--

mcnsifModulationTable OBJECT-TYPE  
SYNTAX SEQUENCE OF McnsifModulationEntry  
MAX-ACCESS not-accessible  
STATUS current  
DESCRIPTION  
"Describes a modulation profile associated with one or more upstream channels."  
::= { mcnsifMIBObjects 6 }

mcnsifModulationEntry OBJECT-TYPE  
SYNTAX McnsifModulationEntry  
MAX-ACCESS not-accessible  
STATUS current  
DESCRIPTION  
"Describes a modulation profile for an Interval Usage Code for one or more upstream channels."  
INDEX { mcnsifModIndex,  
mcnsifModIntervalUsageCode }  
::= { mcnsifModulationTable 1 }

McnsifModulationEntry ::= SEQUENCE {  
mcnsifModIndex Integer32,  
mcnsifModIntervalUsageCode INTEGER,  
mcnsifModControl INTEGER,  
mcnsifModType INTEGER,  
mcnsifModPreambleStart INTEGER,  
mcnsifModPreambleLen INTEGER,

mcnsifModDifferentialEncoding    INTEGER,

Anderson, Sawyer, Woundy

[Page 25]

```
mcnsifModFECErrorCorrection    INTEGER,
mcnsifModFECCodewordLength    INTEGER,
mcnsifModScramblerSeed        INTEGER,
mcnsifModMaxBurstSize         INTEGER,
mcnsifModGuardTimeSize        Integer32,
mcnsifModLastCodeword         INTEGER,
mcnsifModScrambler            INTEGER
}
```

mcnsifModIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An index into the Channel Modulation table representing a group of Interval Usage Codes, all associated with the same channel."

::= { mcnsifModulationEntry 1 }

mcnsifModIntervalUsageCode OBJECT-TYPE

```
SYNTAX      INTEGER {
                request(1),
                requestData(2),
                initialRanging(3),
                periodicRanging(4),
                shortData(5),
                longData(6)
            }
```

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An index into the Channel Modulation table which, when grouped with other Interval Usage Codes, fully instantiate all modulation sets for a given upstream channel."

::= { mcnsifModulationEntry 2 }

mcnsifModControl OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Controls and reflects the status of rows in this table. This object is read-only at the CM."

::= { mcnsifModulationEntry 3 }

mcnsifModType OBJECT-TYPE

```
SYNTAX      INTEGER {
                qpsk(1),
                qam16(2)
            }
```

}

Anderson, Sawyer, Woundy

[Page 26]

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The modulation type used on this channel. This object is read-only at the CM."  
DEFVAL { qpsk } -- if the CMTS supports qpsk  
::= { mcnsifModulationEntry 4 }

mcnsifModPreambleStart OBJECT-TYPE  
SYNTAX INTEGER (0..1023)  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The starting bit offset into mcnsifPreambleValue to be used for the preamble for this modulation profile. This object is read-only at the CM."  
DEFVAL { 0 }  
::= { mcnsifModulationEntry 5 }

mcnsifModPreambleLen OBJECT-TYPE  
SYNTAX INTEGER (0..1024)  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The number of valid bits to be extracted from mcnsifPreambleValue. This object is read-only at the CM."  
-- DEFVAL is the minimum needed by the implementation at the CMTS.  
::= { mcnsifModulationEntry 6 }

mcnsifModDifferentialEncoding OBJECT-TYPE  
SYNTAX INTEGER {  
    on(1),  
    off(2)  
}  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"Whether or not differential encoding is used on this channel. This object is read-only at the CM"  
DEFVAL { off }  
::= { mcnsifModulationEntry 7 }

mcnsifModFECErrorCorrection OBJECT-TYPE  
SYNTAX INTEGER (0..10)  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current



## DESCRIPTION

"The number of correctable errored bytes (t) used in forward error correction code. The value of 0 indicates no correction is employed. The number of check bytes appended will be twice this value. This object is read-only at the CM."

DEFVAL { 0 }

::= { mcnsifModulationEntry 8 }

mcnsifModFECCodewordLength OBJECT-TYPE

SYNTAX INTEGER (1..255)

MAX-ACCESS read-create -- see security note in section 3 of [\[MCNS11\]](#)

STATUS current

## DESCRIPTION

"The number of data bytes (k) in the forward error correction codeword. This object is read-only at the CM "

DEFVAL { 32 } -- note that this is not used unless

mcnsifModFECErrorCorrection is nonzero.

::= { mcnsifModulationEntry 9 }

mcnsifModScramblerSeed OBJECT-TYPE

SYNTAX INTEGER (0..32767)

MAX-ACCESS read-create -- see security note in section 3 of [\[MCNS11\]](#)

STATUS current

## DESCRIPTION

"The 15 bit seed value for the scrambler polynomial. This object is read-only at the CM."

DEFVAL { 0 }

::= { mcnsifModulationEntry 10 }

mcnsifModMaxBurstSize OBJECT-TYPE

SYNTAX INTEGER (0..255)

MAX-ACCESS read-create -- see security note in section 3 of [\[MCNS11\]](#)

STATUS current

## DESCRIPTION

"The maximum number of mini-slots that can be transmitted during this channel's burst time. Returns zero if the burst length is bounded by the allocation MAP rather than this profile. This object is read-only at the CM."

DEFVAL { 0 } -- except for shortData, which is 8.

::= { mcnsifModulationEntry 11 }

mcnsifModGuardTimeSize OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only -- this is dependent on other objects above

STATUS current

## DESCRIPTION

"The number of symbol-times which must follow the end of this



```
channel's burst."  
-- DEFVAL is the minimum time needed by the implementation  
::= { mcsifModulationEntry 12 }
```

mcnsifModLastCodeword OBJECT-TYPE

SYNTAX INTEGER {  
                  fixed(1),  
                  shortened(2)  
                  }

MAX-ACCESS read-create -- this is dependent on other objects  
above

STATUS current

DESCRIPTION

"Whether the last FEC codeword is truncated."

DEFVAL { shortened }

::= { mcnsifModulationEntry 13 }

mcnsifModScrambler OBJECT-TYPE

SYNTAX INTEGER {  
                  on(1),  
                  off(2)  
                  }

MAX-ACCESS read-create -- this is dependent on other objects  
above

STATUS current

DESCRIPTION

"Whether the scrambler is employed."

DEFVAL { off }

::= { mcnsifModulationEntry 14 }

--

-- The Cable Modem Service Table

--

mcnsifCmServiceTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifCmServiceEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes the attributes of each upstream service queue on a  
CM."

::= { mcnsifMIBObjects 7 }

mcnsifCmServiceEntry OBJECT-TYPE

SYNTAX McnsifCmServiceEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes the attributes of an upstream bandwidth service  
queue."

INDEX { ifIndex, mcnsifCmServiceId }

::= { mcnsifCmServiceTable 1 }

```
McnsifCmServiceEntry ::= SEQUENCE {  
    mcnsifCmServiceId          INTEGER,
```

```
mcnsifCmServiceAdminStatus    INTEGER,
mcnsifCmServiceQosProfile     Integer32,
mcnsifCmServiceTxSlotsImmed   Counter32,
mcnsifCmServiceTxSlotsDed     Counter32,
mcnsifCmServiceTxRetries      Counter32,
mcnsifCmServiceTxExceeded     Counter32,
mcnsifCmServiceRqRetries      Counter32,
mcnsifCmServiceRqExceeded     Counter32
}
```

mcnsifCmServiceId            OBJECT-TYPE

SYNTAX            INTEGER (1..8191)

MAX-ACCESS        not-accessible

STATUS    current

DESCRIPTION

"Identifies a service queue for upstream bandwidth. The attributes of this service queue are shared between the CM and the CMTS. The CMTS allocates upstream bandwidth to this service queue based on requests from the CM and on the class of service associated with this queue."

::= { mcnsifCmServiceEntry 1 }

mcnsifCmServiceAdminStatus    OBJECT-TYPE

SYNTAX            INTEGER {  
                  enabled(1),  
                  disabled(2),  
                  destroyed(3) }

MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]

STATUS    current

DESCRIPTION

"Allows a service class for a particular modem to be suppressed, (re-)enabled, or deleted altogether."

::= { mcnsifCmServiceEntry 2 }

mcnsifCmServiceQosProfile      OBJECT-TYPE

SYNTAX    Integer32

MAX-ACCESS        read-only

STATUS    current

DESCRIPTION

"The index in mcnsifQosProfileTable describing the quality of service attributes associated with this particular service."

::= { mcnsifCmServiceEntry 3 }

mcnsifCmServiceTxSlotsImmed    OBJECT-TYPE

SYNTAX            Counter32

MAX-ACCESS        read-only

STATUS    current



## DESCRIPTION

"The number of upstream mini-slots which have been used to transmit data PDUs in immediate (contention) mode. This includes only those PDUs which are presumed to have arrived at the head-end (i.e., those which were explicitly acknowledged.) It does not include retransmission attempts or mini-slots used by Requests."

::= { mcnsifCmServiceEntry 4 }

mcnsifCmServiceTxSlotsDed OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of upstream mini-slots which have been used to transmit data PDUs in dedicated mode (i.e., as a result of a unicast Data Grant)."

::= { mcnsifCmServiceEntry 5 }

mcnsifCmServiceTxRetries OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of attempts to transmit data PDUs containing requests for acknowledgment which did not result in acknowledgment."

::= { mcnsifCmServiceEntry 6 }

mcnsifCmServiceTxExceeded OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of data PDUs transmission failures due to excessive retries without acknowledgment."

::= { mcnsifCmServiceEntry 7 }

mcnsifCmServiceRqRetries OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The number of attempts to transmit bandwidth requests which did not result in acknowledgment."

::= { mcnsifCmServiceEntry 8 }



```
mcnsifCmServiceRqExceeded      OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
"The number of requests for bandwidth which failed due to
excessive retries without acknowledgment."
 ::= { mcnsifCmServiceEntry 9 }

--
-- The CMTS Service Table.
--

mcnsifCmtsServiceTable  OBJECT-TYPE
SYNTAX  SEQUENCE OF McnsifCmtsServiceEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
"Describes the attributes of each upstream service queue on a
CMTS."
 ::= { mcnsifMIBObjects 8 }

mcnsifCmtsServiceEntry  OBJECT-TYPE
SYNTAX  McnsifCmtsServiceEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
"Describes the attributes of an upstream bandwidth service
queue."
INDEX    { ifIndex, mcnsifCmtsServiceId }
 ::= { mcnsifCmtsServiceTable 1 }

McnsifCmtsServiceEntry ::= SEQUENCE {
mcnsifCmtsServiceId      INTEGER,
mcnsifCmtsServiceAdminStatus  INTEGER,
mcnsifCmtsServiceQosProfile  Integer32,
mcnsifCmtsServiceCreateTime  TimeTicks,
mcnsifCmtsServiceCmAddress    MacAddress,
mcnsifCmtsServiceCmIpAddress  IpAddress,
mcnsifCmtsServiceRqDataGrants Counter32,
mcnsifCmtsServiceRqDataSlots  Counter32,
mcnsifCmtsServiceTotalDataGrants    Counter32,
mcnsifCmtsServiceTotalDataSlots     Counter32
}

mcnsifCmtsServiceId      OBJECT-TYPE
SYNTAX  INTEGER (1..8191)
MAX-ACCESS      not-accessible
```



STATUS current

Anderson, Sawyer, Woundy

[Page 32]

## DESCRIPTION

"Identifies a service queue for upstream bandwidth. The attributes of this service queue are shared between the CM and the CMTS. The CMTS allocates upstream bandwidth to this service queue based on requests from the CM and on the class of service associated with this queue."

::= { mcnsifCmtsServiceEntry 1 }

mcnsifCmtsServiceAdminStatus OBJECT-TYPE

SYNTAX INTEGER {  
enabled(1),  
disabled(2),  
destroyed(3) }

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

## DESCRIPTION

"Allows a service class for a particular modem to be suppressed, (re-)enabled, or deleted altogether."

::= { mcnsifCmtsServiceEntry 2 }

mcnsifCmtsServiceQosProfile OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

## DESCRIPTION

"The index in mcnsifQosProfileTable describing the quality of service attributes associated with this particular service."

::= { mcnsifCmtsServiceEntry 3 }

mcnsifCmtsServiceCreateTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The value of sysUpTime when this entry was created."

::= { mcnsifCmtsServiceEntry 4 }

mcnsifCmtsServiceCmAddress OBJECT-TYPE

SYNTAX MacAddress

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

" Identifies the corresponding CM for this Service ID. If the CM is unknown, or if the Service ID is shared among several CMs, returns 00-00-00-00-00-00."

::= { mcnsifCmtsServiceEntry 5 }

mcnsifCmtsServiceCmIpAddress OBJECT-TYPE

SYNTAX IPAddress

Anderson, Sawyer, Woundy

[Page 33]

MAX-ACCESS read-only

STATUS current

DESCRIPTION

" Identifies the corresponding CM for this Service ID. If the CM is unknown, or if the Service ID is shared among several CMs, returns 0.0.0.0."

::= { mcnsifCmtsServiceEntry 6 }

mcnsifCmtsServiceRqDataGrants OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The cumulative number of CMTS unicast Data Grants, as a response to CM upstream bandwidth requests for this Service ID."

::= { mcnsifCmtsServiceEntry 7 }

mcnsifCmtsServiceRqDataSlots OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The cumulative number of mini-slots for CMTS unicast Data Grants, as a response to CM upstream bandwidth requests for this Service ID."

::= { mcnsifCmtsServiceEntry 8 }

mcnsifCmtsServiceTotalDataGrants OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The cumulative number of all CMTS unicast Data Grants for this Service ID."

::= { mcnsifCmtsServiceEntry 9 }

mcnsifCmtsServiceTotalDataSlots OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The cumulative number of mini-slots for all CMTS unicast Data Grants for this Service ID. Also includes any contention (immediate) mini-slots in which the CM successfully transmitted data."

::= { mcnsifCmtsServiceEntry 10 }



```
-- The following table describes the attributes of each class of
service.
-- The entries in this table are referenced from the
mcnsifServiceEntries.
-- They exist as a separate table in order to reduce redundant
information
-- in mcnsifServiceTable.
--
-- This table is implemented at both the CM and the CMTS. The
significance
-- of mcnsifQosProfIndex is local to the device. The CM need only
maintain
-- entries for the classes of service referenced by its
mcnsifServiceTable.
--
```

```
mcnsifQosProfileTable OBJECT-TYPE
SYNTAX SEQUENCE OF McnsifQosProfileEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Describes the attributes of each class of service."
 ::= { mcnsifMIBObjects 9 }
```

```
mcnsifQosProfileEntry OBJECT-TYPE
SYNTAX McnsifQosProfileEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Describes the attributes of a particular class of service."
INDEX { mcnsifQosProfIndex }
 ::= { mcnsifQosProfileTable 1 }
```

```
McnsifQosProfileEntry ::= SEQUENCE {
mcnsifQosProfIndex Integer32,
mcnsifQosProfPriority INTEGER,
mcnsifQosProfMaxUpBandwidth INTEGER,
mcnsifQosProfGuarUpBandwidth INTEGER,
mcnsifQosProfMaxDownBandwidth INTEGER,
mcnsifQosProfMaxTxBurst Integer32
}
```

```
mcnsifQosProfIndex OBJECT-TYPE
SYNTAX Integer32
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Of significance only to the local device. An index must appear
```

```
if it is referenced in the mcsifServiceTable of the device."  
::= { mcsifQosProfileEntry 1 }
```

mcnsifQosProfPriority OBJECT-TYPE

SYNTAX INTEGER (0..7)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"A relative priority assigned to this service when allocating bandwidth. Zero indicates lowest priority; seven highest. Interpretation of priority is device-specific."

::= { mcnsifQosProfileEntry 2 }

mcnsifQosProfMaxUpBandwidth OBJECT-TYPE

SYNTAX INTEGER (0..100000000)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum upstream bandwidth, in bits per second, allowed for a service with this service class. Zero if there is no restriction of upstream bandwidth."

::= { mcnsifQosProfileEntry 3 }

mcnsifQosProfGuarUpBandwidth OBJECT-TYPE

SYNTAX INTEGER (0..100000000)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Minimum guaranteed upstream bandwidth, in bits per second, allowed for a service with this service class."

::= { mcnsifQosProfileEntry 4 }

mcnsifQosProfMaxDownBandwidth OBJECT-TYPE

SYNTAX INTEGER (0..400000000)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum downstream bandwidth, in bits per second, allowed for a service with this service class. Zero if there is no restriction of downstream bandwidth."

::= { mcnsifQosProfileEntry 5 }

mcnsifQosProfMaxTxBurst OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The maximum number of mini-slots that may be requested for a single upstream transmission."

::= { mcnsifQosProfileEntry 6 }





--

-- This table is implemented only at the CMTS. The table is optional,  
-- but if implemented all objects within it must be implemented.

--

mcnsifCmtsTallyTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifCmtsTallyEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"For the MCNS MAC layer, this group maintains a number of statistical tallies related to the MAC protocol itself. This group is implemented only at the CMTS."

::= { mcnsifMIBObjects 10 }

mcnsifCmtsTallyEntry OBJECT-TYPE

SYNTAX McnsifCmtsTallyEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"For the MCNS MAC layer, this group maintains a set of statistical tallies, either by MSAP or by both MSAP and channel. If an ifEntry exists for an upstream or downstream channel then an entry for that channel should appear in this table as well."

INDEX { ifIndex }

::= { mcnsifCmtsTallyTable 1 }

McnsifCmtsTallyEntry ::= SEQUENCE {

mcnsifTxSyncs Counter32,

mcnsifTxUcds Counter32,

mcnsifTxMaps Counter32,

mcnsifRxRangeRequests Counter32,

mcnsifTxRangeResponses Counter32,

mcnsifRxRegRequests Counter32,

mcnsifTxRegResponses Counter32,

mcnsifTxUpChannelChangeReqs Counter32,

mcnsifRxUpChannelChangeRsps Counter32

}

mcnsifTxSyncs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of MAC SYNC messages transmitted."

::= { mcnsifCmtsTallyEntry 1 }

mcnsifTxUcds OBJECT-TYPE



MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Upstream Channel Descriptor messages  
transmitted."  
::= { mcnsifCmtsTallyEntry 2 }

mcnsifTxMaps OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC allocation Map messages transmitted."  
::= { mcnsifCmtsTallyEntry 3 }

mcnsifRxRangeRequests OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Ranging Request messages received."  
::= { mcnsifCmtsTallyEntry 4 }

mcnsifTxRangeResponses OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Ranging Response messages transmitted."  
::= { mcnsifCmtsTallyEntry 5 }

mcnsifRxRegRequests OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Registration Requests received."  
::= { mcnsifCmtsTallyEntry 6 }

mcnsifTxRegResponses OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Registration Responses transmitted."  
::= { mcnsifCmtsTallyEntry 7 }



mcnsifTxUpChannelChangeReqs      OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of MAC Upstream Channel Change Requests transmitted."

::= { mcnsifCmtsTallyEntry 8 }

mcnsifRxUpChannelChangeRsps      OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of MAC Upstream Channel Change Responses received."

::= { mcnsifCmtsTallyEntry 9 }

--

-- This table is implemented only at the CM. The table is optional,

-- but if implemented all objects within it must be implemented.

--

mcnsifCmTallyTable              OBJECT-TYPE

SYNTAX SEQUENCE OF McnsifCmTallyEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"For the MCNS MAC layer, this group maintains a number of statistical tallies related to the MAC protocol itself. This group is implemented only at the CM."

::= { mcnsifMIBObjects 11 }

mcnsifCmTallyEntry              OBJECT-TYPE

SYNTAX McnsifCmTallyEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"For the MCNS MAC layer, this group maintains a set of statistical tallies, either by MSAP or by both MSAP and channel."

INDEX { ifIndex }

::= { mcnsifCmTallyTable 1 }

McnsifCmTallyEntry ::= SEQUENCE {

mcnsifRxSyncs                  Counter32,

mcnsifRxUcDs                  Counter32,

mcnsifRxMaps                  Counter32,

mcnsifTxRangeRequests        Counter32,

mcnsifRxRangeResponses       Counter32,



```
mcnsifTxRegRequests      Counter32,  
mcnsifRxRegResponses     Counter32,  
mcnsifRxUpChannelChangeReqs Counter32,  
mcnsifTxUpChannelChangeRsps Counter32  
}
```

```
mcnsifRxSyncs            OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC SYNC messages received."  
::= { mcnsifCmTallyEntry 1 }
```

```
mcnsifRxUcds             OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Upstream Channel Descriptor messages received."  
::= { mcnsifCmTallyEntry 2 }
```

```
mcnsifRxMaps             OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC allocation Map messages received."  
::= { mcnsifCmTallyEntry 3 }
```

```
mcnsifTxRangeRequests    OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Ranging Request messages transmitted."  
::= { mcnsifCmTallyEntry 4 }
```

```
mcnsifRxRangeResponses   OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Ranging Responses received."  
::= { mcnsifCmTallyEntry 5 }
```

```
mcnsifTxRegRequests      OBJECT-TYPE  
SYNTAX Counter32
```





MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Registration Request messages transmitted."  
::= { mcnsifCmTallyEntry 6 }

mcnsifRxRegResponses OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Registration Response messages received."  
::= { mcnsifCmTallyEntry 7 }

mcnsifRxUpChannelChangeReqs OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Upstream Channel Change Requests received."  
::= { mcnsifCmTallyEntry 8 }

mcnsifTxUpChannelChangeRsps OBJECT-TYPE  
SYNTAX Counter32  
MAX-ACCESS read-only  
STATUS current  
DESCRIPTION  
"Number of MAC Upstream Channel Change Responses transmitted."  
::= { mcnsifCmTallyEntry 9 }

--  
-- The following table is implemented on the CM and the CMTS.  
--

mcnsifSignalQualityTable OBJECT-TYPE  
SYNTAX SEQUENCE OF McnsifSignalQualityEntry  
MAX-ACCESS not-accessible  
STATUS current  
DESCRIPTION  
"At the CM, describes the PHY signal quality of downstream channels. At the CMTS, describes the PHY signal quality of upstream channels. At the CMTS, this table may exclude contention intervals."  
::= { mcnsifMIBObjects 12 }

mcnsifSignalQualityEntry OBJECT-TYPE  
SYNTAX McnsifSignalQualityEntry  
MAX-ACCESS not-accessible



STATUS current

DESCRIPTION

"At the CM, describes the PHY characteristics of a downstream channel. At the CMTS, describes the PHY signal quality of an upstream channel."

INDEX { ifIndex }

::= { mcnsifSignalQualityTable 1 }

```
McnsifSignalQualityEntry ::= SEQUENCE {
    mcnsifSigQIncludesContention    INTEGER,
    mcnsifSigQUnerroreds           Counter32,
    mcnsifSigQCorrecteds           Counter32,
    mcnsifSigQUncorrectables       Counter32,
    mcnsifSigQSignalNoise          Integer32,
    mcnsifSigQRxPowerLevel         INTEGER,
    mcnsifSigQInChannelResponse    INTEGER,
    mcnsifSigQMicroreflections     INTEGER
}
```

mcnsifSigQIncludesContention OBJECT-TYPE

SYNTAX INTEGER { true(1), false(2) }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"True if this CMTS includes contention intervals in the counters in this table. Always false(2) for CMs."

::= { mcnsifSignalQualityEntry 1 }

mcnsifSigQUnerroreds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of codewords received on this channel without error. This includes all codewords, whether or not they were part of frames destined for this device."

::= { mcnsifSignalQualityEntry 2 }

mcnsifSigQCorrecteds OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of codewords received on this channel with correctable errors. This includes all codewords, whether or not they were part of frames destined for this device."

::= { mcnsifSignalQualityEntry 3 }



mcnsifSigQUncorrectables OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of codewords received on this channel with uncorrectable errors. This includes all codewords, whether or not they were part of frames destined for this CM."

::= { mcnsifSignalQualityEntry 4 }

mcnsifSigQSignalNoise OBJECT-TYPE

SYNTAX TenthdBmV

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Signal/Noise ratio as perceived at this."

::= { mcnsifSignalQualityEntry 5 }

mcnsifSigQRxPowerLevel OBJECT-TYPE

SYNTAX TenthdBmV

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Indicates the power level of the received signal. Only meaningful at the CM."

::= { mcnsifSignalQualityEntry 6 }

mcnsifSigQInChannelResponse OBJECT-TYPE

SYNTAX INTEGER {  
    other(1),  
    excellent(2),  
    good(3),  
    ok(4),  
    fair(5),  
    poor(6),  
    unknown(7)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Provides a rough indication of the received in-channel response."

::= { mcnsifSignalQualityEntry 7 }

mcnsifSigQMicroreflections OBJECT-TYPE

SYNTAX INTEGER {  
    other(1),  
    excellent(2),  
    good(3),  
}



```
        ok(4),
        fair(5),
        poor(6),
        unknown(7)
    }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Provides a rough indication of the (lack of) micro-
    reflections."
 ::= { mcnsifSignalQualityEntry 8 }

mcnsifConformance OBJECT IDENTIFIER ::= { mcnsifMIB 2 }
mcnsifGroups      OBJECT IDENTIFIER ::= { mcnsifConformance 1 }
mcnsifCompliances OBJECT IDENTIFIER ::= { mcnsifConformance 2 }

mcnsifBasicCompliance MODULE-COMPLIANCE
STATUS          current
DESCRIPTION
    "The compliance statement for devices that implement the MCNS
    Radio Frequency Interfaces."
MODULE -- mcnsifMIB
MANDATORY-GROUPS {mcnsifDownstreamChannelGroup,
    mcnsifUpstreamChannelGroup,
    mcnsifPreambleGroup,
    mcnsifModulationGroup,
    mcnsifQosProfileGroup,
    mcnsifSignalQualityGroup }
GROUP mcnsifCmMacGroup
    DESCRIPTION
    "This group is implemented only on Cable Modems, not on CMTSS."
GROUP mcnsifCmServiceGroup
    DESCRIPTION
    "This group is implemented only on Cable Modems, not on CMTSS."
GROUP mcnsifCmtsMacGroup
    DESCRIPTION
    "This group is implemented only on CMTSS, not on CMs."
GROUP mcnsifCmtsServiceGroup
    DESCRIPTION
    "This group is implemented only on CMTSS, not on CMs."
OBJECT mcnsifDownChannelFrequency -- within mcnsifDownstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
    "Read-write at the CMTS; read-only at CMs."
OBJECT mcnsifDownChannelModulation -- within mcnsifDownstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
    "Read-write at the CMTS; read-only at CMs."
```





```
OBJECT mcnSifDownChannelInterleave    -- within mcnSifDownstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelFrequency        -- within mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelSymbolRate      -- within mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelModulationProfile -- within
mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelSlotSize        -- within mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelDownChannelTiming -- within
mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelRangingBackoffStart -- within
mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelRangingBackoffEnd -- within
mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelTxBackoffStart    -- within mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifUpChannelTxBackoffEnd      -- within mcnSifUpstreamChannelGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
OBJECT mcnSifPreambleValue              -- within mcnSifPreambleGroup
MIN-ACCESS read-only
DESCRIPTION
"Read-write at the CMTS; read-only at CMs."
```



OBJECT mcnsifModControl -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModType -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModPreambleStart -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModPreambleLen -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModDifferentialEncoding -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModFECErrorCorrection -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModFECCodewordLength -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModScramblerSeed -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModMaxBurstSize -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModLastCodeword -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifModScrambler -- within mcnsifModulationGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Read-create at the CMTS; read-only at CMs."  
OBJECT mcnsifCmServiceAdminStatus -- within mcnsifCmServiceGroup  
MIN-ACCESS read-only  
DESCRIPTION  
"Only read-only access is required."



OBJECT mcnsifCmtsServiceAdminStatus -- within mcnsifCmtsServiceGroup  
MIN-ACCESS read-only

## DESCRIPTION

"Only read-only access is required."

OBJECT mcnsifCmtsServiceQosProfile -- within mcnsifCmtsServiceGroup

MIN-ACCESS read-only

## DESCRIPTION

"Only read-only access is required."

::= { mcnsifCompliances 1 }

mcnsifCmMacGroup OBJECT-GROUP

OBJECTS { mcnsifCmtsAddress,  
          mcnsifCmCapabilities,  
          mcnsifCmResets,  
          mcnsifCmRangingRespTimeout }

STATUS current

## DESCRIPTION

"At the CM, a collection of objects describing attributes of each MAC entry, extending the information in ifEntry."

::= { mcnsifGroups 1 }

mcnsifCmtsMacGroup OBJECT-GROUP

OBJECTS { mcnsifCmtsCapabilities,  
          mcnsifSyncInterval,  
          mcnsifUcdInterval,  
          mcnsifMinMapSize,  
          mcnsifMaxMapSize,  
          mcnsifMaxFutureMap,  
          mcnsifMaxServiceIds,  
          mcnsifRqPerMap,  
          mcnsifDataPerMap,  
          mcnsifInsertionInterval,  
          mcnsifInvitedRangingRetries }

STATUS current

## DESCRIPTION

"At the CMTS, a collection of objects describing attributes of each MAC entry, extending the information in ifEntry."

::= { mcnsifGroups 2 }

mcnsifDownstreamChannelGroup OBJECT-GROUP

OBJECTS { mcnsifDownChannelId,  
          mcnsifDownChannelFrequency,  
          mcnsifDownChannelModulation,  
          mcnsifDownChannelInterleave,  
          mcnsifDownChannelTxPower }

STATUS current

## DESCRIPTION

"At the CMTS, describes the attributes of each downstream



channel (frequency band). At the CM, describes the attached downstream channel."

::= { mcnSifGroups 3 }

mcnsifUpstreamChannelGroup OBJECT-GROUP

OBJECTS { mcnsifUpChannelId,  
          mcnsifUpChannelFrequency,  
          mcnsifUpChannelSymbolRate,  
          mcnsifUpChannelModulationProfile,  
          mcnsifUpChannelSlotSize,  
          mcnsifUpChannelTxPower,  
          mcnsifUpChannelDownChannelTiming,  
          mcnsifUpChannelTxTimingOffset,  
          mcnsifUpChannelRangingBackoffStart,  
          mcnsifUpChannelRangingBackoffEnd,  
          mcnsifUpChannelTxBackoffStart,  
          mcnsifUpChannelTxBackoffEnd }

STATUS current

DESCRIPTION

"At the CMTS, describes the attributes of each upstream channel (frequency band). At the CM, describes the attached upstream channel."

::= { mcnSifGroups 4 }

mcnsifPreambleGroup OBJECT-GROUP

OBJECTS { mcnsifPreambleValue }

STATUS current

DESCRIPTION

"Describes a baseline preamble string for one or more upstream channels."

::= { mcnSifGroups 5 }

mcnsifModulationGroup OBJECT-GROUP

OBJECTS { mcnsifModType,  
          mcnsifModPreambleStart,  
          mcnsifModPreambleLen,  
          mcnsifModDifferentialEncoding,  
          mcnsifModFECErrorCorrection,  
          mcnsifModFECCodewordLength,  
          mcnsifModScramblerSeed,  
          mcnsifModMaxBurstSize,  
          mcnsifModGuardTimeSize,  
          mcnsifModLastCodeword,  
          mcnsifModScrambler }

STATUS current

DESCRIPTION

"Describes a modulation profile for an Interval Usage Code for one or more upstream channels."

::= { mcnSifGroups 6 }





## mcnsifCmServiceGroup OBJECT-GROUP

```
OBJECTS { mcnsifCmServiceAdminStatus,
          mcnsifCmServiceQosProfile,
          mcnsifCmServiceTxSlotsImmed,
          mcnsifCmServiceTxSlotsDed,
          mcnsifCmServiceTxRetries,
          mcnsifCmServiceTxExceeded,
          mcnsifCmServiceRqRetries,
          mcnsifCmServiceRqExceeded }
```

STATUS current

## DESCRIPTION

"At the CM, describes the attributes of an upstream bandwidth service queue."

::= { mcnsifGroups 7 }

## mcnsifCmtsServiceGroup OBJECT-GROUP

```
OBJECTS { mcnsifCmtsServiceAdminStatus,
          mcnsifCmtsServiceQosProfile,
          mcnsifCmtsServiceCreateTime,
          mcnsifCmtsServiceCmAddress,
          mcnsifCmtsServiceCmIpAddress,
          mcnsifCmtsServiceRqDataGrants,
          mcnsifCmtsServiceRqDataSlots,
          mcnsifCmtsServiceTotalDataGrants,
          mcnsifCmtsServiceTotalDataSlots }
```

STATUS current

## DESCRIPTION

"At the CMTS, describes the attributes of an upstream bandwidth service queue."

::= { mcnsifGroups 8 }

## mcnsifQosProfileGroup OBJECT-GROUP

```
OBJECTS { mcnsifQosProfPriority,
          mcnsifQosProfMaxUpBandwidth,
          mcnsifQosProfGuarUpBandwidth,
          mcnsifQosProfMaxDownBandwidth,
          mcnsifQosProfMaxTxBurst }
```

STATUS current

## DESCRIPTION

"Describes the attributes of a particular class of service."

::= { mcnsifGroups 9 }

## mcnsifCmtsTallyGroup OBJECT-GROUP -- This group is purely optional

```
OBJECTS { mcnsifTxSyncs,
          mcnsifTxUcds,
          mcnsifTxMaps,
          mcnsifRxRangeRequests,
          mcnsifTxRangeResponses,
```



```
    mcnsifRxRegRequests,  
    mcnsifTxRegResponses,  
    mcnsifTxUpChannelChangeReqs,  
    mcnsifRxUpChannelChangeRsps }
```

STATUS current

DESCRIPTION

"For the MCNS MAC layer, this group maintains a set of statistical tallies, either by MSAP or by both MSAP and channel. If an ifEntry exists for an upstream or downstream channel then an entry for that channel should appear in this table as well."  
 ::= { mcnsifGroups 10 }

mcnsifCmTallyGroup OBJECT-GROUP -- This group is purely optional

```
OBJECTS { mcnsifRxSyncs,  
    mcnsifRxUcds,  
    mcnsifRxMaps,  
    mcnsifTxRangeRequests,  
    mcnsifRxRangeResponses,  
    mcnsifTxRegRequests,  
    mcnsifRxRegResponses,  
    mcnsifRxUpChannelChangeReqs,  
    mcnsifTxUpChannelChangeRsps }
```

STATUS current

DESCRIPTION

"For the MCNS MAC layer, this group maintains a set of statistical tallies, either by MSAP or by both MSAP and channel."  
 ::= { mcnsifGroups 11 }

mcnsifSignalQualityGroup OBJECT-GROUP

```
OBJECTS { mcnsifSigQIncludesContention,  
    mcnsifSigQUnerroreds,  
    mcnsifSigQCorrecteds,  
    mcnsifSigQUncorrectables,  
    mcnsifSigQSignalNoise,  
    mcnsifSigQRxPowerLevel,  
    mcnsifSigQInChannelResponse,  
    mcnsifSigQMicroreflections }
```

STATUS current

DESCRIPTION

"At the CM, describes the PHY characteristics of a downstream channel. At the CMTS, describes the PHY signal quality of an upstream channel."  
 ::= { mcnsifGroups 12 }

END

### **5.3 Definitions of Managed Objects for Cable Modems and Cable Modem Termination Systems**



The following groups of objects are provided for management of CMs and CMTSs:

-mcnscmBase - extends the MIB-II 'system' group with objects needed for cable modem system management.

-mcnscmNmAccessTable - provides a minimum level of SNMP access security (see Section 3 of [[MCNS11](#)]).

-mcnscmSoftware - provides information for network-downloadable software upgrades

-mcnscmServer - provides information about the progress of the interaction with various provisioning servers.

-mcnscmEvent - provides control and logging for event reporting

-mcnscmFilter - configures filters at link layer and IP layer for bridged data traffic.

CableModem-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY, OBJECT-TYPE,  
IpAddress, Integer32, Counter32, TimeTicks  
FROM SNMPv2-SMI

DisplayString, RowStatus, DateAndTime  
FROM SNMPv2-TC

OBJECT-GROUP, MODULE-COMPLIANCE  
FROM SNMPv2-CONF

mcns  
FROM MCNS-MIB

;

mcnscmMIB MODULE-IDENTITY

LAST-UPDATED "9703241130Z"

ORGANIZATION "Multimedia Cable Network System"

CONTACT-INFO " Pam Anderson  
Postal: CableLabs  
400 Centennial Parkway  
Louisville, CO 80027-1266  
Tel: +1 303 661 9100  
Fax: +1 303 661 9199  
E-mail: p.anderson@cablelabs.com"

DESCRIPTION

"This is the MIB Module for MCNS-compliant cable modems and



cable-modem termination systems."

::= { mcns 2 }

mcnscmMIBObjects OBJECT IDENTIFIER ::= { mcnscmMIB 1 }

mcnscmBase OBJECT IDENTIFIER ::= { mcnscmMIBObjects 1 }

--

-- For the following object, there is no concept in the  
-- RFI specification corresponding to a backup CMTS. The  
-- enumeration is provided here in case someone is able  
-- to define such a role or device.

--

mcnscmRole OBJECT-TYPE

SYNTAX INTEGER {  
    cm(1),  
    cmtsActive(2),  
    cmtsBackup(3)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Defines the current role of this device."

::= { mcnscmBase 1 }

--

-- This object is mandatory in the CM and optional in the CMTS

--

mcnscmDateTime OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The date and time, with optional timezone information."

::= { mcnscmBase 2 }

mcnscmResetNow OBJECT-TYPE

SYNTAX INTEGER {  
    reset(1),  
    ready(2)  
}

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Setting this object to reset(1) causes the device to reset."

::= { mcnscmBase 3 }





mcnscmSerialNumber        OBJECT-TYPE

SYNTAX    DisplayString

MAX-ACCESS        read-only

STATUS    current

DESCRIPTION

"The manufacturer's serial number for this device."

::= { mcnscmBase 4 }

mcnscmSTPControl        OBJECT-TYPE

SYNTAX    INTEGER {  
                  stEnabled(1),  
                  noStFilterBpdu(2),  
                  noStPassBpdu(3)  
                  }

MAX-ACCESS        read-write        -- see security note in section 3 of [[MCNS11](#)]

STATUS    current

DESCRIPTION

"This object controls operation of the spanning tree protocol (as distinguished from transparent bridging). If set to st-enabled(1) then the spanning tree protocol is enabled, subject to bridging constraints. If no-st-filter-bpdu(2), then spanning tree is not active, and Bridge PDUs received are discarded. If no-st-pass-bpdu(3) then spanning tree is not active and Bridge PDUs are transparently forwarded. Note that a device need not implement all of these options, but that no-st-filter-bpdu(2) is required."

::= { mcnscmBase 5 }

--

-- The following table provides one level of security for access to the  
-- device by network management stations. Also, see the security section  
-- in the text above. Note also that access is also constrained by the  
-- community strings and any vendor-specific security.

--

mcnscmNmAccessTable        OBJECT-TYPE

SYNTAX    SEQUENCE OF McnscmNmAccessEntry

MAX-ACCESS        not-accessible

STATUS    current

DESCRIPTION

"Controls access to SNMP objects by network management stations."

::= { mcnscmMIBObjects 2 }

mcnscmNmAccessEntry        OBJECT-TYPE

SYNTAX    McnscmNmAccessEntry

MAX-ACCESS        not-accessible

STATUS                      current

Anderson, Sawyer, Woundy

[Page 53]

## DESCRIPTION

"Controls access to SNMP objects by a particular network management station. For each row-entry in this table, the contents are not readable unless the management station has read-write permission."

```
INDEX    { mcnscmNmAccessIndex  }  
::= {    mcnscmNmAccessTable 1 }
```

```
McnscmNmAccessEntry ::= SEQUENCE {  
mcnscmNmAccessIndex      Integer32,  
mcnscmNmAccessIp         IPAddress,  
mcnscmNmAccessIpMask     IPAddress,  
mcnscmNmAccessCommunity  DisplayString,  
mcnscmNmAccessControl    INTEGER,  
mcnscmNmAccessInterfaces OCTET STRING  
}
```

mcnscmNmAccessIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"Index used to order the application of access entries."  
::= { mcnscmNmAccessEntry 1 }

mcnscmNmAccessIp OBJECT-TYPE

SYNTAX IPAddress

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The IP address (or subnet) of the network management station. The address 255.255.255.255 is defined to mean any NMS. If traps are enabled for this entry, then the value must be the address of a specific device."  
DEFVAL { 'ffffffff'h }  
::= { mcnscmNmAccessEntry 2 }

mcnscmNmAccessIpMask OBJECT-TYPE

SYNTAX IPAddress

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The IP subnet mask of the network management stations. If traps are enabled for this entry, then the value must be 255.255.255.255."  
DEFVAL { 'ffffffff'h }  
::= { mcnscmNmAccessEntry 3 }



mcnscmNmAccessCommunity            OBJECT-TYPE  
SYNTAX            OCTET STRING (SIZE (0..127))  
MAX-ACCESS        read-create  
STATUS            current

## DESCRIPTION

"The community string to be matched for access by this entry. If set to the null string then any community string will match."

DEFVAL { "public" }

::= { mcnscmNmAccessEntry 4 }

mcnscmNmAccessControl    OBJECT-TYPE  
SYNTAX    INTEGER {

    none(1),  
    read(2),  
    readWrite(3),  
    roWithTraps(4),  
    rwWithTraps(5),  
    trapsOnly(6)  
  }

MAX-ACCESS        read-create        -- see security note in section 3 of [[MCNS11](#)]

STATUS    current

## DESCRIPTION

"Specifies the type of access allowed to this NMS. Setting this object to none(1) causes the table entry to be destroyed. Read (2) allows access by 'get' and 'get-next' PDUs. ReadWrite(3) allows access by 'set' as well. RoWithtraps(4), rwWithTraps(5), and trapsOnly(6) control distribution of Trap PDUs transmitted by this device."

DEFVAL { read }

::= { mcnscmNmAccessEntry 5 }

-- The syntax of the following object was copied from [RFC1493](#),  
-- dot1dStaticAllowedToGoTo.

mcnscmNmAccessInterfaces    OBJECT-TYPE  
SYNTAX            OCTET STRING

MAX-ACCESS        read-create        -- see security note in section 3 of [[MCNS11](#)]

STATUS    current

## DESCRIPTION

"Specifies the set of interfaces from which requests from this NMS will be accepted.

Each octet within the value of this object specifies a set of eight interfaces, with the first octet specifying ports 1 through 8, the second octet specifying interfaces 9

through 16, etc. Within each octet, the most significant bit represents the lowest

numbered interface, and the least significant bit represents the

highest numbered

Anderson, Sawyer, Woundy

[Page 55]

interface. Thus, each interface is represented by a single bit within the value of this object. If that bit has a value of '1' then that interface is included in the set.

Note that entries in this table apply only to link-layer interfaces (e.g., Ethernet and CATV MAC). Upstream and downstream channel interfaces may not be specified."

-- DEFVAL is the bitmask corresponding to all interfaces  
::= { mcnscmNmAccessEntry 6 }

--

-- Procedures for using the following group are described in [section 4.2.2](#)

--

mcnscmSoftware OBJECT IDENTIFIER ::= { mcnscmMIBObjects 3 }

mcnscmSwServer OBJECT-TYPE

SYNTAX IpAddress

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The address of the TFTP server used for software upgrades."

::= { mcnscmSoftware 1 }

mcnscmSwFilename OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..32))

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The file name of the software image to be loaded into this device. Unless set via SNMP, this is the file name specified by the provisioning server that corresponds to the software version that is desired for this device. If unknown, the string '(unknown)' is returned."

::= { mcnscmSoftware 2 }

mcnscmSwAdminStatus OBJECT-TYPE

SYNTAX INTEGER {

upgradeFromMgt(1),

allowProvisioningUpgrade(2)

}

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"If set to upgradeFromMgt(1), the device initiates a TFTP



download using mcnscmSwFilename. On successfully receiving an

Anderson, Sawyer, Woundy

[Page 56]

image, the device will re-boot. On reboots, the device will disregard software information from the provisioning server. If set to allowProvisioningUpgrade(2), the device will use the software version information supplied by the provisioning server when next rebooting (this does not cause a re-boot). "

::= { mcnscmSoftware 3 }

mcnscmSwOperStatus OBJECT-TYPE

SYNTAX INTEGER {  
    inProgress(1),  
    completeFromProvisioning(2),  
    completeFromMgt(3),  
    failed(4),  
    other(5)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"InProgress(1) indicates that a TFTP download is underway, either as a result of a version mismatch at provisioning or as a result of a upgradeFromMgt request. CompleteFromProvisioning(2) indicates that the last software upgrade was a result of version mismatch at provisioning. CompleteFromMgt(3) indicates that the last software upgrade was a result of setting mcnscmSwAdminStatus to upgradeFromMgt. Failed(4) indicates that the last attempted download failed, ordinarily due to TFTP timeout."

::= { mcnscmSoftware 4 }

--

-- The following group describes server access and parameters used for  
-- initial provisioning and bootstrapping.

--

mcnscmServer OBJECT IDENTIFIER ::= { mcnscmMIBObjects 4 }

mcnscmServerBootState OBJECT-TYPE

SYNTAX INTEGER {  
    operational(1),  
    disabled(2),  
    waitingForDhcpOffer(3),  
    waitingForDhcpResponse(4),  
    waitingForTimeServer(5),  
    waitingForSecurity(6),  
    waitingForTftp(7),  
    refusedByCmts(8),  
    forwardingDenied(9),  
    other(10),



```
        unknown(11)
    }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "If operational(1), the device has completed loading and
    processing of configuration
    parameters and the CMTS has completed the Registration exchange.
    If disabled(2)
    then the device was administratively disabled, possibly by being
    refused network
    access in the configuration file. If waitingForDhcpOffer(3) then
    a DHCP Discover has
    been transmitted and no offer has yet been received. If
    waitingForDhcpResponse(4)
    then a DHCP Request has been transmitted and no response has yet
    been received. If
    waitingForSecurity(5) then an exchange with a security server is
    needed and has not
    completed. If waitingForTftp(6) then a request to the TFTP
    parameter server has been
    made and no response received. If refusedByCmts(7) then the
    Registration
    Request/Response exchange with the CMTS failed. If
    forwardingDenied(8) then the
    registration process completed, but the network access option in
    the received
    configuration file prohibits forwarding. "
 ::= { mcnscmServer 1 }
```

```
mcnscmServerDhcp      OBJECT-TYPE
SYNTAX  IpAddress
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The IP address of the DHCP server that assigned an IP address
    to this device. Returns 0.0.0.0 if DHCP was not used for IP
    address assignment."
 ::= { mcnscmServer 2 }
```

```
mcnscmServerTime      OBJECT-TYPE
SYNTAX  IpAddress
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The IP address of the Time server (RFC-868)."
 ::= { mcnscmServer 3 }
```



mcnscmServerTftp OBJECT-TYPE

SYNTAX IPAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The IP address of the TFTP server responsible for downloading provisioning and configuration parameters to this device."

::= { mcnscmServer 4 }

mcnscmServerSecurity OBJECT-TYPE

SYNTAX IPAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The IP address of the security server, if present."

::= { mcnscmServer 5 }

mcnscmServerConfigFile OBJECT-TYPE

SYNTAX DisplayString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The name of the device configuration file read from the TFTP server."

::= { mcnscmServer 6 }

--

-- Event Reporting

-- Note that this group has implications for enterprise-specific trap reporting. See [section 4.3](#) for details.

--

mcnscmEvent OBJECT IDENTIFIER ::= { mcnscmMIBObjects 5 }

mcnscmEvControl OBJECT-TYPE

SYNTAX INTEGER {  
resetLog(1),  
useDefaultReporting(2)  
}

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Setting this object to resetLog(1) empties the event log. All data is deleted. Setting it to useDefaultReporting(2) returns all event priorities to their factory-default reporting."

::= { mcnscmEvent 1 }

mcnscmEvSyslog OBJECT-TYPE

SYNTAX

IpAddress

Anderson, Sawyer, Woundy

[Page 59]

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]  
STATUS current

## DESCRIPTION

"The IP address of the Syslog server. If 0.0.0.0, syslog transmission is inhibited."

::= { mcnscmEvent 2 }

mcnscmEvThrottleAdminStatus OBJECT-TYPE

SYNTAX INTEGER {  
    unconstrained(1),  
    maintainBelowThreshold(2),  
    stopAtThreshold(3),  
    inhibited(4)  
}

MAX-ACCESS read-write

STATUS current

## DESCRIPTION

"Controls the transmission of traps and syslog messages with respect to the trap pacing threshold. unconstrained(1) causes traps to be transmitted without regard to the threshold settings. maintainBelowThreshold(2) causes trap transmission to be suppressed if the number of traps would otherwise exceed the threshold. stopAtThreshold(3) causes trap transmission to cease at the threshold, and not resume until directed to do so. inhibited(4) causes all trap transmission to be suppressed.

Writing to this object resets the thresholding state."

::= { mcnscmEvent 3 }

mcnscmEvThrottleOperStatus OBJECT-TYPE

SYNTAX INTEGER {  
    inhibited(1),  
    operational(2)  
}

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"Indicates whether or not trap and syslog transmission is currently inhibited due to thresholds and mcnscmEvThrottleAdminStatus."

::= { mcnscmEvent 4 }

mcnscmEvThrottleThreshold OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-write

STATUS current





## DESCRIPTION

"Number of trap/syslog events per mcnscmEvThrottleInterval to be transmitted before throttling."

::= { mcnscmEvent 5 }

mcnscmEvThrottleInterval OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-write

STATUS current

## DESCRIPTION

"The interval over which the trap threshold applies."

::= { mcnscmEvent 6 }

--

-- The following table controls the reporting of the various classes of events. For each event priority,  
-- a combination of logging and reporting mechanisms may be chosen. The mapping of event types  
-- to priorities is vendor-dependent. Vendors may also choose to allow the user to control that mapping  
-- through proprietary means.

mcnscmEvControlTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnscmEvControlEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"Allows control of the reporting of event classes."

::= { mcnscmEvent 7 }

mcnscmEvControlEntry OBJECT-TYPE

SYNTAX McnscmEvControlEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"Allows configuration of the reporting mechanisms for a particular event priority."

INDEX { mcnscmEvPriority }

::= { mcnscmEvControlTable 1 }

McnscmEvControlEntry ::= SEQUENCE {

mcnscmEvPriority INTEGER,

mcnscmEvReporting INTEGER

}



mcnscmEvPriority OBJECT-TYPE

SYNTAX INTEGER {  
    emergency(1),  
    alert(2),  
    critical(3),  
    error(4),  
    warning(5),  
    notice(6),  
    information(7),  
    debug(8)  
}

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The priority level that is controlled by this entry."

::= { mcnscmEvControlEntry 1 }

mcnscmEvReporting OBJECT-TYPE

SYNTAX INTEGER {  
    noReporting(1),  
    localLogOnly(2),  
    trapOnly(3),  
    localLogAndTrap(4),  
    syslogOnly(5),  
    bothLogs(6),  
    syslogAndTrap(7),  
    bothLogsAndTrap(8)  
}

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Defines the action to be taken on occurrence of this event class. Implementations may not necessarily support all options for all event classes, but at minimum must allow traps and syslogging to be disabled."

::= { mcnscmEvControlEntry 2 }

mcnscmEventTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnscmEventEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Contains a log of network and device events that may be of interest in fault isolation and troubleshooting."

::= { mcnscmEvent 8 }

mcnscmEventEntry OBJECT-TYPE

SYNTAX McnscmEventEntry

MAX-ACCESS not-accessible

Anderson, Sawyer, Woundy

[Page 62]

STATUS current

DESCRIPTION

"Describes a network or device event that may be of interest in fault isolation and troubleshooting."

INDEX { mcnscmEvIndex }  
 ::= { mcnscmEventTable 1 }

McnscmEventEntry ::=

SEQUENCE {  
 mcnscmEvIndex Integer32,  
 mcnscmEvFirstTime DateAndTime,  
 mcnscmEvLastTime DateAndTime,  
 mcnscmEvCount Counter32,  
 mcnscmEvLevel INTEGER,  
 mcnscmEvId Integer32,  
 mcnscmEvText DisplayString  
 }

mcnscmEvIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Provides relative ordering of the objects in the event log. This object will always increase except when (a) the log is reset via mcnscmEvControl, (b) the device reboots and does not implement nonvolatile storage for this log, or (c) it reaches the value 2<sup>31</sup>. In all three cases the log is cleared and the next entry is 1."

::= { mcnscmEventEntry 1 }

mcnscmEvFirstTime OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The time that this entry was created."

::= { mcnscmEventEntry 2 }

mcnscmEvLastTime OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If multiple events are reported via the same entry, the time that the last event for this entry occurred."

::= { mcnscmEventEntry 3 }



mcnscmEvCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of consecutive event instances reported by this entry."

::= { mcnscmEventEntry 4 }

mcnscmEvLevel OBJECT-TYPE

SYNTAX INTEGER {  
    emergency(1),  
    alert(2),  
    critical(3),  
    error(4),  
    warning(5),  
    notice(6),  
    information(7),  
    debug(8)  
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The priority level of this event."

::= { mcnscmEventEntry 5 }

--

-- Vendors will provide their own enumerations for the following.

-- The interpretation of the enumeration is unambiguous for a particular

-- value of the vendor's enterprise number in sysObjectID.

--

mcnscmEvId OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"For this product, uniquely identifies the type of event that is reported by this entry."

::= { mcnscmEventEntry 6 }

mcnscmEvText OBJECT-TYPE

SYNTAX DisplayString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Provides a human-readable description of the event, including all relevant context (interface numbers, etc.)"



```
::= { mcnscmEventEntry 7 }
```

mcnscmFilter OBJECT IDENTIFIER ::= { mcnscmMIBObjects 6 }

-- LLC filters can be defined on an inclusive or exclusive basis: CMs  
can be configured to forward only  
-- packets matching a set of layer three protocols, or to drop packets  
matching a set of layer three protocols.  
-- CMs must support a minimum of ten LLC filter entries.

mcnscmFilterLLCDefault OBJECT-TYPE

SYNTAX INTEGER {  
                  discard(1),  
                  accept(2)  
                  }

MAX-ACCESS read-write -- see security note in section 3 of [\[MCNS11\]](#)

STATUS current

DESCRIPTION

"If set to discard(1), all packets matching an LLC filter will  
be discarded. If set to accept(2), only packets matching an LLC  
filter will be accepted for further processing (e.g.,  
bridging)."

::= { mcnscmFilter 1 }

mcnscmFilterLLCTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnscmFilterLLCEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A list of filters to apply to (bridged) LLC traffic, which  
forwards or drops packets on the basis of the layer three  
protocol type."

::= { mcnscmFilter 2 }

mcnscmFilterLLCEntry OBJECT-TYPE

SYNTAX McnscmFilterLLCEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes a filter to apply to (bridged) LLC traffic received  
on a specified interface. "

INDEX { mcnscmFilterLLCIndex }

::= { mcnscmFilterLLCTable 1 }

McnscmFilterLLCEntry ::= SEQUENCE {

mcnscmFilterLLCIndex	Integer32,
mcnscmFilterLLCControl	INTEGER,
mcnscmFilterLLCIfIndex	Integer32,
mcnscmFilterLLCProtocolType	INTEGER,

```
mcnscmFilterLLCProtocol  
mcnscmFilterLLCMatches  
}
```

```
Integer32,  
Counter32
```

mcnscmFilterLLCIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index used for the identification of filters (note that LLC filter order is irrelevant)."

::= { mcnscmFilterLLCEntry 1 }

mcnscmFilterLLCControl OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Controls and reflects the status of rows in this table."

::= { mcnscmFilterLLCEntry 2 }

mcnscmFilterLLCIfIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The entry interface to which this filter applies. The value corresponds to ifIndex for either a CATV MAC or other network interface."

-- DEFVAL the customer-side interface for CMs; undefined for CMTSS

::= { mcnscmFilterLLCEntry 3 }

mcnscmFilterLLCProtocolType OBJECT-TYPE

SYNTAX INTEGER {  
    ethertype(1),  
    dsap(2)  
}

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The format of the value in mcnscmFilterLLCProtocol: either a two-byte Ethernet Ethertype, or a one-byte 802.2 SAP value. EtherType(1) also applies to SNAP-encapsulated frames."

DEFVAL { ethertype }

::= { mcnscmFilterLLCEntry 4 }

mcnscmFilterLLCProtocol OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION



"The layer three protocol for which this filter applies. The protocol value format depends on mcnscmFilterLLCProtocolType. Note that for SNAP frames, etherType filtering is performed rather than DSAP=0xAA."

DEFVAL { 0 }

::= { mcnscmFilterLLCEntry 5 }

mcnscmFilterLLCMatches OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Counts the number of times this filter was matched."

::= { mcnscmFilterLLCEntry 6 }

-- The default behavior for (bridged) packets that do not match IP filters is defined by

-- mcnscmFilterIpDefault.

mcnscmFilterIpDefault OBJECT-TYPE

SYNTAX INTEGER {  
discard(1),  
accept(2)  
}

MAX-ACCESS read-write -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"If set to discard(1), all packets not matching an IP filter will be discarded. If set to accept(2), all packets not matching an IP filter will be accepted for further processing (e.g., bridging)."

::= { mcnscmFilter 3 }

--

-- CMs must support a minimum of ten IP address filter entries.

--

mcnscmFilterIpTable OBJECT-TYPE

SYNTAX SEQUENCE OF McnscmFilterIpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An ordered list of filters to apply to IP traffic. Filter application is ordered by the filter index, rather than by a best match algorithm. Packets which match no filters are discarded or forwarded according to the setting of mcnscmFilterIpDefault."

::= { mcnscmFilter 4 }



mcnscmFilterIpEntry OBJECT-TYPE

SYNTAX McnscmFilterIpEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Describes a filter to apply to IP traffic received on a specified interface. Both source and destination addresses must match for the filter to apply."

INDEX { mcnscmFilterIpIndex }

::= { mcnscmFilterIpTable 1 }

McnscmFilterIpEntry ::= SEQUENCE {

    mcnscmFilterIpIndex Integer32,  
    mcnscmFilterIpControl INTEGER,  
    mcnscmFilterIpIfIndex Integer32,  
    mcnscmFilterIpDirection INTEGER,  
    mcnscmFilterIpBroadcast INTEGER,  
    mcnscmFilterIpSaddr IpAddress,  
    mcnscmFilterIpSmask IpAddress,  
    mcnscmFilterIpDaddr IpAddress,  
    mcnscmFilterIpDmask IpAddress,  
    mcnscmFilterIpProtocol INTEGER,  
    mcnscmFilterIpSourcePortLow Integer32,  
    mcnscmFilterIpSourcePortHigh Integer32,  
    mcnscmFilterIpDestPortLow Integer32,  
    mcnscmFilterIpDestPortHigh Integer32,  
    mcnscmFilterIpMatches Counter32  
}

mcnscmFilterIpIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index used to order the application of filters."

::= { mcnscmFilterIpEntry 1 }

mcnscmFilterIpStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Controls and reflects the status of rows in this table."

::= { mcnscmFilterIpEntry 2 }

mcnscmFilterIpControl OBJECT-TYPE

SYNTAX INTEGER {

    discard(1),



```
accept(2)  
}
```

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"If set to discard(1), all packets matching this filter will be discarded and scanning of the remainder of the filter list will be aborted. If set to accept(2), all packets matching this filter will be accepted for further processing (e.g., bridging) and scanning of the remainder of the filter list will be aborted. "

DEFVAL { discard }

::= { mcnscmFilterIpEntry 3 }

mcnscmFilterIpIfIndex OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"The entry interface to which this filter applies. The value corresponds to ifIndex for either a CATV MAC or other network interface."

--DEFVAL for CMs is the customer-side (i.e., ethernet) interface. Not defined for CMTSS.

::= { mcnscmFilterIpEntry 4 }

mcnscmFilterIpDirection OBJECT-TYPE

SYNTAX INTEGER {  
    inbound(1),  
    outbound(2)  
}

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Determines whether the filter is applied to inbound(1) or outbound(2) traffic."

DEFVAL { inbound }

::= { mcnscmFilterIpEntry 5 }

mcnscmFilterIpBroadcast OBJECT-TYPE

SYNTAX INTEGER {  
    broadcastOnly(1),  
    allPackets(2)  
}

MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]

STATUS current

DESCRIPTION

"Determines whether the filter is applied to all traffic, or just to multicast and broadcast traffic."

DEFVAL { allPackets }

```
::= { mcns cmFilterIpEntry 6 }
```

mcnscmFilterIpSaddr        OBJECT-TYPE  
SYNTAX    IpAddress  
MAX-ACCESS        read-create        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"The source IP address, or portion thereof, that is to be  
matched for this filter."  
DEFVAL { '00000000'h }  
::= { mcnscmFilterIpEntry 7 }

mcnscmFilterIpSmask        OBJECT-TYPE  
SYNTAX    IpAddress  
MAX-ACCESS        read-create        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
"A bit mask that is to be applied to the source address prior to  
matching. This mask is not necessarily the same as a subnet  
mask, but 1's bits must be leftmost and contiguous."  
DEFVAL { '00000000'h } ::= { mcnscmFilterIpEntry 8 }

mcnscmFilterIpDaddr        OBJECT-TYPE  
SYNTAX    IpAddress  
MAX-ACCESS        read-create        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
" The destination IP address, or portion thereof, that is to be  
matched for this filter "  
DEFVAL { '00000000'h }  
::= { mcnscmFilterIpEntry 9 }

mcnscmFilterIpDmask        OBJECT-TYPE  
SYNTAX    IpAddress  
MAX-ACCESS        read-create        -- see security note in section 3 of [[MCNS11](#)]  
STATUS    current  
DESCRIPTION  
" A bit mask that is to be applied to the destination address  
prior to matching. This mask is not necessarily the same as a  
subnet mask, but 1's bits must be leftmost and contiguous "  
DEFVAL { '00000000'h }  
::= { mcnscmFilterIpEntry 10 }

mcnscmFilterIpProtocol    OBJECT-TYPE  
SYNTAX    INTEGER {  
          icmp(1),  
          tcp(6),  
          udp(17),  
          any(256)  
          }  
          }



MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"The IP protocol value that is to be matched."  
DEFVAL { any }  
::= { mcnscmFilterIpEntry 11 }

mcnscmFilterIpSourcePortLow OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"If mcnscmFilterIpProtocol is udp or tcp, this is the inclusive lower bound of the transport-layer source port range that is to be matched."  
DEFVAL { 0 }  
::= { mcnscmFilterIpEntry 12 }

mcnscmFilterIpSourcePortHigh OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"If mcnscmFilterIpProtocol is udp or tcp, this is the inclusive upper bound of the transport-layer source port range that is to be matched."  
DEFVAL { 65535 }  
::= { mcnscmFilterIpEntry 13 }

mcnscmFilterIpDestPortLow OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"If mcnscmFilterIpProtocol is udp or tcp, this is the inclusive lower bound of the transport-layer destination port range that is to be matched."  
DEFVAL { 0 }  
::= { mcnscmFilterIpEntry 14 }

mcnscmFilterIpDestPortHigh OBJECT-TYPE  
SYNTAX Integer32  
MAX-ACCESS read-create -- see security note in section 3 of [[MCNS11](#)]  
STATUS current  
DESCRIPTION  
"If mcnscmFilterIpProtocol is udp or tcp, this is the inclusive upper bound of the transport-layer destination port range that is to be matched."

```
DEFVAL { 65535 }  
::= { mcnscmFilterIpEntry 15 }
```

mcnscmFilterIpMatches OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Counts the number of times this filter was matched."

::= { mcnscmFilterIpEntry 16 }

mcnscmConformance OBJECT IDENTIFIER ::= { mcnscmMIB 2 }

mcnscmGroups OBJECT IDENTIFIER ::= { mcnscmConformance 1 }

mcnscmCompliances OBJECT IDENTIFIER ::= { mcnscmConformance 2 }

mcnscmBasicCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"The compliance statement for MCNS Cable Modems and Cable Modem Termination Systems."

MODULE -- mcnscmMIB

MANDATORY-GROUPS {mcnscmBaseGroup,  
mcnscmEventGroup, mcnscmEvControlGroup,  
mcnscmEvGroup, mcnscmFilterGroup, mcnscmFilterLLCGroup,  
mcnscmFilterIpGroup }

GROUP mcnscmNmAccessGroup

DESCRIPTION

"Must be implemented on Cable Modems, should be implemented on CMTSs"

GROUP mcnscmServerGroup

DESCRIPTION

"This group is implemented only on Cable Modems, not on CMTSs."

GROUP mcnscmSoftwareGroup

DESCRIPTION

"Must be implemented on Cable Modems, should be implemented on CMTSs."

OBJECT mcnscmSTPControl

MIN-ACCESS read-only

DESCRIPTION

"At minimum, devices need only support no-st-filter-bpdu(2)."

::= { mcnscmCompliances 1 }

mcnscmBaseGroup OBJECT-GROUP

OBJECTS { mcnscmRole,  
mcnscmDateTime,  
mcnscmResetNow,  
mcnscmSerialNumber,  
mcnscmSTPControl }

STATUS current

DESCRIPTION

"A collection of objects providing device status and control."

::= { mcnscmGroups 1 }





## mcnscmNmAccessGroup OBJECT-GROUP

OBJECTS { mcnscmNmAccessIp,  
          mcnscmNmAccessIpMask,  
          mcnscmNmAccessCommunity,  
          mcnscmNmAccessControl,  
          mcnscmNmAccessInterfaces }

STATUS current

## DESCRIPTION

"A collection of objects for controlling access to SNMP objects."

::= { mcnscmGroups 2 }

## mcnscmSoftwareGroup OBJECT-GROUP

OBJECTS { mcnscmSwServer,  
          mcnscmSwFilename,  
          mcnscmSwAdminStatus,  
          mcnscmSwOperStatus }

STATUS current

## DESCRIPTION

"A collection of objects for controlling software downloads."

::= { mcnscmGroups 3 }

## mcnscmServerGroup OBJECT-GROUP

OBJECTS { mcnscmServerBootState,  
          mcnscmServerDhcp,  
          mcnscmServerTime,  
          mcnscmServerTftp,  
          mcnscmServerSecurity,  
          mcnscmServerConfigFile }

STATUS current

## DESCRIPTION

"A collection of objects providing status about server provisioning."

::= { mcnscmGroups 4 }

## mcnscmEventGroup OBJECT-GROUP

OBJECTS { mcnscmEvControl,  
          mcnscmEvSyslog,  
          mcnscmEvThrottleAdminStatus,  
          mcnscmEvThrottleOperStatus,  
          mcnscmEvThrottleThreshold,  
          mcnscmEvThrottleInterval }

STATUS current

## DESCRIPTION

"A collection of objects used to control and monitor events."

::= { mcnscmGroups 5 }



mcnscmEvControlGroup OBJECT-GROUP

OBJECTS { mcnscmEvReporting }

STATUS current

DESCRIPTION

"A collection of objects allowing configuration of the reporting mechanisms for a particular event priority."

::= { mcnscmGroups 6 }

mcnscmEvGroup OBJECT-GROUP

OBJECTS { mcnscmEvFirstTime,  
          mcnscmEvLastTime,  
          mcnscmEvCount,  
          mcnscmEvLevel,  
          mcnscmEvId,  
          mcnscmEvText }

STATUS current

DESCRIPTION

"A collection of objects describing a network or device event that may be of interest in fault isolation and troubleshooting."

::= { mcnscmGroups 7 }

mcnscmFilterGroup OBJECT-GROUP

OBJECTS { mcnscmFilterLLCDefault,  
          mcnscmFilterIpDefault }

STATUS current

DESCRIPTION

"Objects controlling default behavior in the event that no filters match."

::= { mcnscmGroups 8 }

mcnscmFilterLLCGroup OBJECT-GROUP

OBJECTS { mcnscmFilterLLCControl,  
          mcnscmFilterLLCIfIndex,  
          mcnscmFilterLLCProtocolType,  
          mcnscmFilterLLCProtocol,  
          mcnscmFilterLLCMatches }

STATUS current

DESCRIPTION

"A collection of objects for filtering at the LLC layer."

::= { mcnscmGroups 9 }

mcnscmFilterIpGroup OBJECT-GROUP

OBJECTS { mcnscmFilterIpControl,  
          mcnscmFilterIpIfIndex,  
          mcnscmFilterIpDirection,  
          mcnscmFilterIpBroadcast,  
          mcnscmFilterIpSaddr,  
          mcnscmFilterIpSmask,



```
        mcnscmFilterIpDaddr,
        mcnscmFilterIpDmask,
        mcnscmFilterIpProtocol,
        mcnscmFilterIpSourcePortLow,
        mcnscmFilterIpSourcePortHigh,
        mcnscmFilterIpDestPortLow,
        mcnscmFilterIpDestPortHigh,
        mcnscmFilterIpMatches }
STATUS   current
DESCRIPTION
"A collection of objects for filtering at the IP layer."
::= { mcnscmGroups 10 }

END
```

## **6. Protocol Definition for SYSLOG**

This section documents the usage of the SYSLOG protocol for the Operations Support System environment. The SYSLOG protocol is a UDP-based protocol that permits remote logging of devices. Messages may be associated with different facilities and multiple priorities.

The basic format of the SYSLOG packet is simple to describe. The UDP source and destination port number is 514. The UDP payload consists of a facility/priority value enclosed in angle brackets, followed by a null-terminated string. The UDP payload string normally includes an optional time-of-day stamp, an identification string, an optional PID (in square brackets), and the actual logging message.

For consistency in a multi-vendor CM environment, this section adds further constraints to the SYSLOG packet. The CM uses the "local0" facility in its SYSLOG messages, so that the SYSLOG server can manage CM SYSLOG messages separately from kernel, mail, news, and other generic facilities. This limits the facility/priority values to the range of 128 to 135. The actual facility/priority value depends on the urgency of the message: emergency(128), alert(129), critical(130), error(131), warning(132), notice(133), information(134), and debug(135).

This section also constrains the UDP payload string. The time-of-day stamp SHOULD NOT be included, forcing the SYSLOG server to provide its own (consistent) timestamps for all CM SYSLOG messages. The identification string MUST be "Cablemodem", and the "optional PID" MUST be a constant vendor-specific identification label, to assist in SYSLOG server logging management.

An example of a valid SYSLOG UDP payload would be "<132>Cablemodem[VendorX]: Downloading new CM software". This example

payload might be recorded on the SYSLOG server as "Jan 12 12:56:03  
[24.1.1.1](#) Cablemodem[VendorX]: Downloading new CM software".

## 7. References

[CSMIMIB] Masuma Ahmed and Mario P. Vecchi. Definitions of Managed Objects for HFC RF Spectrum Management (internet draft [draft-ahmed-csmimib-mib-00.txt](#)). June 13, 1996

[MCNS1] MCNS Data Over Cable Service Interface Specification Request for Proposals, December 11, 1995 (can be downloaded on the World Wide Web from <http://www.cablemodem.com>)

[MCNS3] MCNS Cable Modem Termination System - Network-Side Interface Specification SP-CMTS-NSID04-960409 (CMTS-NSI), April 9, 1996

[MCNS4] MCNS Cable Modem to Customer Premise Equipment Interface Specification SP-CMCID04-960409 (CMCI), April 9, 1996

[MCNS5] MCNS Operations Support System Framework TR-OSSF (under preparation)

[MCNS6] MCNS Data Over Cable Services Cable Modem TELCO Return Interface Specification SP-CMTRID02-961016 (under preparation)

[MCNS7] MCNS Data Over Cable Services Cable Modem Radio Frequency Interface Specification SP-RFID01- 970326

[MCNS8] MCNS Data Over Cable Services Security Specification SP-DOCSS (under preparation)

[MCNS9] Data Over Cable Services Cable Modem to Subscriber Security Module Interface Specification (under preparation)

[MCNS10] CNS Data Over Cable Services Data Privacy Specification (under preparation)

[MCNS11] Data Over Cable Services Operations Support System Interface Specification SP-OSSII01-970403

[MCNS12] Data Over Cable Services Operations Support System Interface Specification Radio Frequency Management Information Base SP-OSSI-RFI01-970403

[MCNS13] Data Over Cable Services Operations Support System Interface Specification Telephony Return Management Information Base SP-OSSI-TR (under preparation)

[MCNS14] Data Over Cable Services Operations Support System Interface Specification Security Management Information Base SP-OSSI-SEC (under preparation)



[RFC-1157] Schoffstall, M., Fedor, M., Davin, J. and Case, J., A Simple Network Management Protocol (SNMP), IETF [RFC-1157](#), May, 1990

Anderson, Sawyer, Woundy

[Page 76]

[RFC-1212] K. McCloghrie and M. Rose. Concise MIB Definitions, IETF RFC-1212, March, 1991

[RFC-1213] K. McCloghrie and M. Rose. Management Information Base for Network Management of TCP/IP-base internets: MIB-II, IETF [RFC-1213](#), March, 1991

[RFC-1224] L. Steinberg., Techniques for Managing Asynchronously Generated Alerts, IETF [RFC-1224](#), May, 1991

[RFC-1493] E. Decker, P. Langille, A. Rijssinghani, and K. McCloghrie., Definitions of Managed Objects for Bridges, IETF [RFC-1493](#), July, 1993

[RFC-1533] S. Alexander and R. Droms. DHCP Options and BOOTP Vendor Extensions. IETF [RFC-1533](#). October, 1993.

[RFC-1573] K. McCloghrie and Kastenholz., Evolution of the Interfaces Group of MIB-II, IETF [RFC-1573](#), January, 1994

[RFC-1643] F. Kastenholz., Definitions of Managed Objects for the Ethernet-like Interface Types, IETF [RFC-1643](#), July, 1994

## **8. Authors' Addresses**

Pamela Anderson  
CableLabs  
[400 Centennial Parkway](#)  
Louisville, CO 80027  
Phone: 303-661-9100  
Fax: 303-661-9199  
Email: p.anderson@cablelabs.com

Wilson Sawyer  
Bay Networks  
Lancity Cable Modem Division  
[200 Bulfinch Drive](#)  
Andover, MA 01810-1140  
Phone: 508-682-1600x240  
Fax: 508-682-3200  
Email: wsawyer@lancity.com

Richard Woundy  
Continental Cablevision  
The Pilot House - Lewis Wharf  
Boston, MA 02110  
Phone: 617-854-3351  
Fax: 617-854-3925

Email: [rwoundy@continental.com](mailto:rwoundy@continental.com)

Anderson, Sawyer, Woundy

[Page 77]