Data-Over-Cable Service Interface Specifications
DOCSIS® 4.0

Cable Modem Operations Support System Interface Specification

CM-SP-CM-OSSIv4.0-I09-231012

ISSUED

DISCLAIMER

This document is furnished on an "AS IS" basis and neither CableLabs nor its members provides any representation or warranty, express or implied, regarding the accuracy, completeness, noninfringement, or fitness for a particular purpose of this document, or any document referenced herein. Any use or reliance on the information or opinion in this document is at the risk of the user, and CableLabs and its members shall not be liable for any damage or injury incurred by any person arising out of the completeness, accuracy, or utility of any information or opinion contained in the document.

CableLabs reserves the right to revise this document for any reason including, but not limited to, changes in laws, regulations, or standards promulgated by various entities, technology advances, or changes in equipment design, manufacturing techniques, or operating procedures described, or referred to, herein.

This document is not to be construed to suggest that any company modify or change any of its products or procedures, nor does this document represent a commitment by CableLabs or any of its members to purchase any product whether or not it meets the characteristics described in the document. Unless granted in a separate written agreement from CableLabs, nothing contained herein shall be construed to confer any license or right to any intellectual property. This document is not to be construed as an endorsement of any product or company or as the adoption or promulgation of any guidelines, standards, or recommendations.

Document Status Sheet

Document Control Number:	CM-SP-CM-OSSIv4.0-I09-231012			
Document Title:	Cable Modem Operations Support System Interface Specification			
Revision History:	D01 – Released 6/28/2019 I01 – Released 8/15/2019 I02 – Released 3/11/2020 I03 – Released 1/27/2021 I04 – Released 5/21/2021 I05 – Released 9/27/2021 I06 – Released 3/2/2022 I07 – Released 11/16/2022 I08 – Released 5/16/2023 I09 – Released 10/12/2023			
Date:	October 12, 2023			
Status:	~~Work in Progress~~	~~Draft~~	**Issued**	~~**Closed**~~
Distribution Restrictions:	~~Author Only~~	~~CL/Member~~	~~CL/ Member/ Vendor~~	**Public**

Key to Document Status Codes

Work in Progress	An incomplete document, designed to guide discussion and generate feedback that may include several alternative requirements for consideration.
Draft	A document in specification format that is considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
Issued	A generally public document that has undergone rigorous Member and Technology Supplier review, cross-vendor interoperability, and is suitable for certification/qualification testing if applicable. Issued Specifications are subject to the Engineering Change Process.
Closed	A static document, reviewed, tested, validated, and closed to further engineering change requests to the specification through CableLabs.

Trademarks

Table of Contents

List of Figures

List of Tables

1 SCOPE

1.1 Introduction and Purpose

This specification is part of the DOCSIS family of specifications developed by Cable Television Laboratories (CableLabs). In particular, this specification is part of a series of specifications that defines the sixth generation of high-speed data-over-cable systems. This specification was developed for the benefit of the cable industry and includes contributions by operators and vendors from North America, Europe, and other regions.

This specification defines the Operations Support System Interface (OSSI) requirements for the Cable Modem (CM).

1.2 Background

1.2.1 Broadband Access Network

A coaxial-based broadband access network is assumed. This may take the form of either an all-coax or hybrid-fiber/coax (HFC) network. The generic term "cable network" is used here to cover all cases.

A cable network uses a tree-and-branch architecture with analog transmission. The key functional characteristics assumed in this document are the following:

- Two-way transmission.
- A maximum optical/electrical spacing between the CMTS and the most distant CM of 100 miles (160 km) in each direction, although typical maximum separation may be 10-15 miles (16-24 km).

1.2.2 Network and System Architecture

1.2.2.1 The DOCSIS Network

The elements that participate in the provisioning of DOCSIS services are shown in Figure 1.

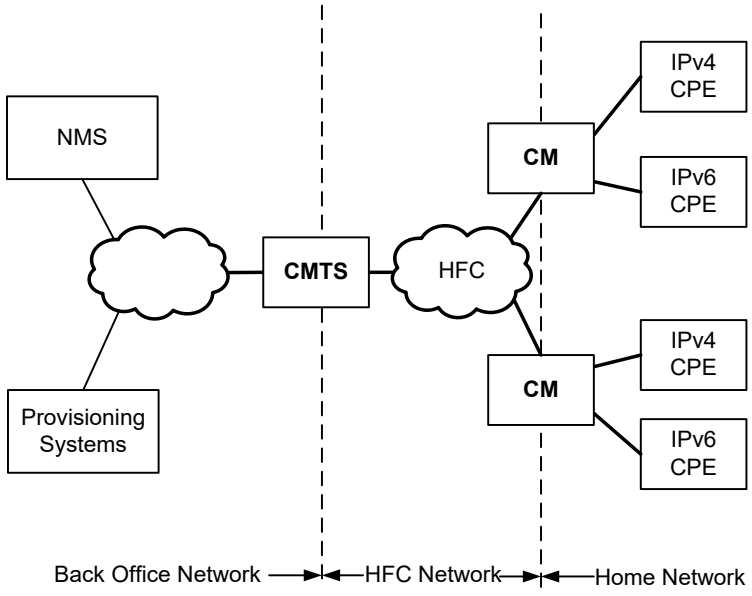


Figure 1 - The DOCSIS Network

The CM connects to the operator's HFC network and to a home network, bridging packets between them. Many CPE devices can connect to the CM's LAN interfaces. CPE devices can be embedded with the CM in a single device, or they can be separate standalone devices (as shown in Figure 1).

CPE devices may use IPv4, IPv6 or both forms of IP addressing. Examples of typical CPE devices are home routers, set-top devices, and personal computers.

The CMTS connects the operator's back office and core network with the HFC network. Its main function is to forward packets between these two domains, and optionally to forward packets between upstream and downstream channels on the HFC network. The CMTS performs this forwarding with any combination of link-layer (bridging) and network-layer (routing) semantics.

Various applications are used to provide back office configuration and other support to the devices on the DOCSIS network. These applications use IPv4 and/or IPv6 as appropriate to the particular operator's deployment. The following applications include:

- Provisioning Systems
  - The DHCP servers provide the CM with initial configuration information, including the device IP address(es), when the CM boots.
  - The Configuration File server is used to download configuration files to CMs when they boot. Configuration files are in binary format and permit the configuration of the CM's parameters.

    The Software Download server is used to download software upgrades to the CM.
  - The Time Protocol server provides Time Protocol clients, typically CMs, with the current time of day.
- Network Management System (NMS)
  - The SNMP Manager allows the operator to configure and monitor SNMP Agents which reside within the Cable Modems.
  - The syslog server collects messages pertaining to the operation of devices.

1.2.3 Service Goals

As cable operators have widely deployed high-speed data services on cable television systems, the demand for bandwidth has increased. Additionally, networks have scaled to such a degree that IPv4 address constraints are becoming a burden on network operations. To this end, CableLabs' member companies added new features to the DOCSIS® specification for the purpose of increasing channel capacity, enhancing network security, expanding addressability of network elements, and deploying new service offerings.

The DOCSIS system allows transparent bi-directional transfer of Internet Protocol (IP) traffic, between the cable system headend and customer locations, over an all-coaxial or hybrid-fiber/coax (HFC) cable network. This is shown in simplified form in Figure 2.

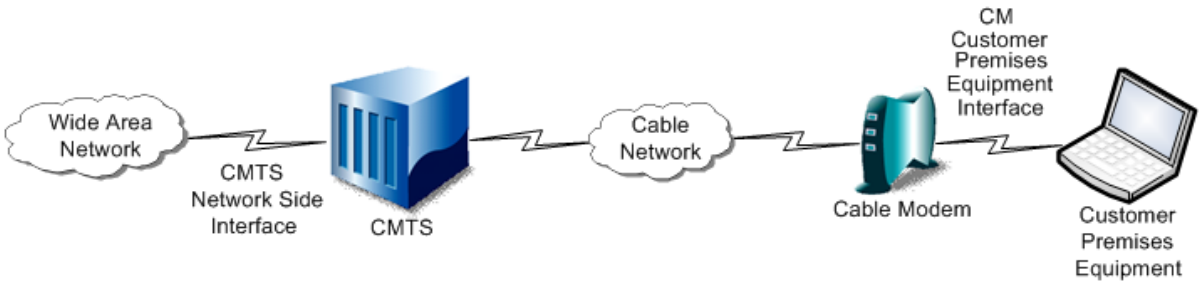


Figure 2 - Transparent IP Traffic through the Data-Over-Cable System

1.2.4 Statement of Compatibility

This specification defines the DOCSIS 4.0 interface. Prior generations of DOCSIS were commonly referred to as DOCSIS 1.0, 1.1, 2.0, 3.0 and 3.1 interfaces. DOCSIS 4.0 is backward-compatible with equipment built to the previous specifications with the exception of DOCSIS 1.0 CMs. DOCSIS 4.0-compliant CMs interoperate seamlessly with DOCSIS 4.0, DOCSIS 3.1, and DOCSIS 3.0 CMTSs. DOCSIS 4.0-compliant CMTSs seamlessly support DOCSIS 3.1, DOCSIS 3.0, DOCSIS 2.0, and DOCSIS 1.1.

1.2.5 Reference Architecture

The reference architecture for data-over-cable services and interfaces is shown in Figure 3.

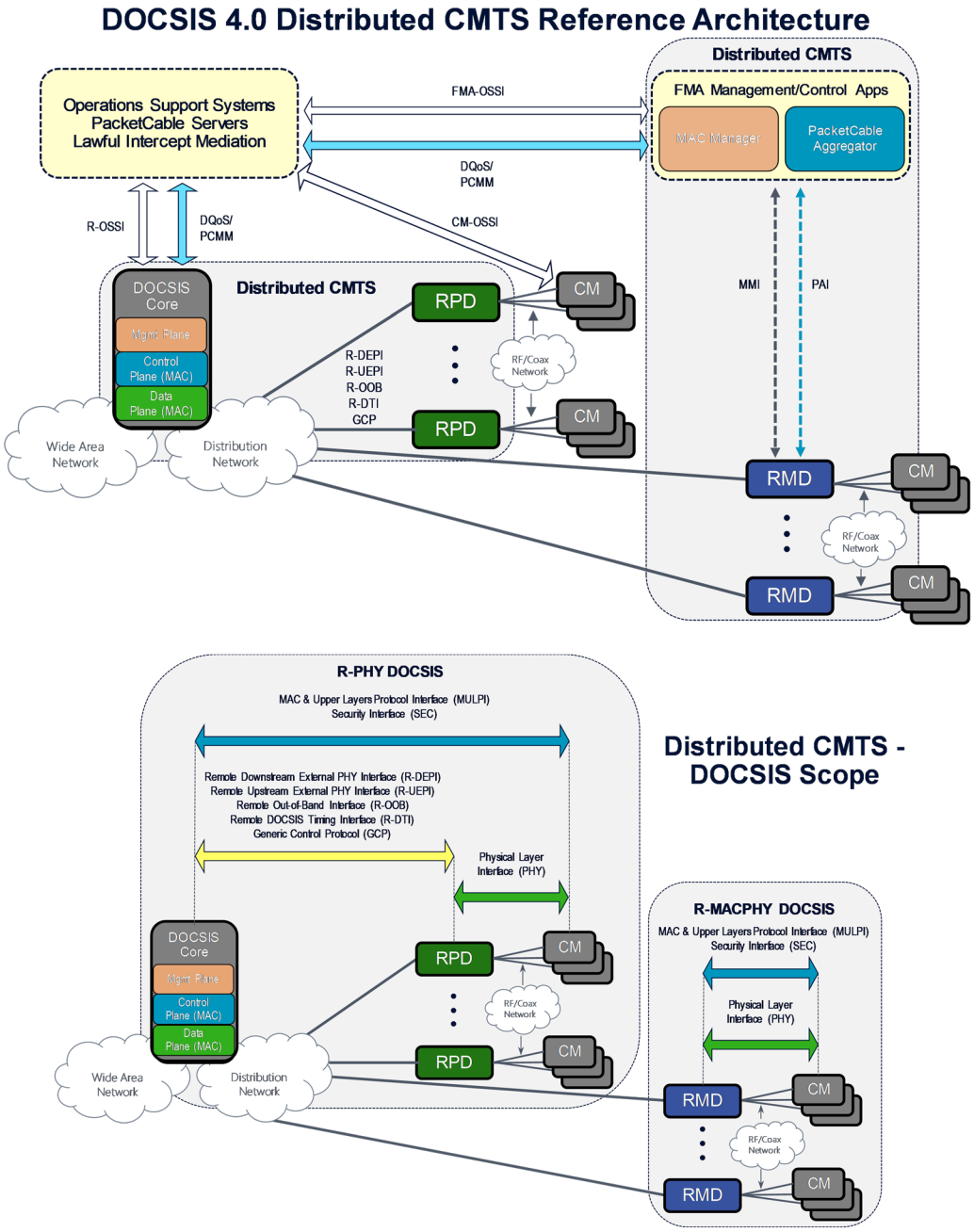


Figure 3 - Data-Over-Cable Reference Architecture

1.2.6 DOCSIS 4.0 Documents

A list of the specifications in the DOCSIS 4.0 series is provided in Table 1. For further information, please refer to http://www.cablemodem.com.

Table 1 - DOCSIS 4.0 Series of Specifications

Designation	Title
CM-SP-PHYv4.0	Physical Layer Specification
CM-SP-MULPIv4.0	Media Access Control and Upper Layer Protocols Interface Specification
CM-SP-CM-OSSIv4.0	Cable Modem Operations Support System Interface Specification
CM-SP-CCAP-OSSIv4.0	Converged Cable Access Platform Operations Support System Interface Specification
CM-SP-SECv4.0	Security Specification
CM-SP-CMCIv3.0	Cable Modem CPE Interface Specification

This specification is defining the interface for the Operations Support Systems Interface (OSSI), specifically for the Cable Modem.

Related DOCSIS specifications are listed in Table 2.

Table 2 - DOCSIS 4.0 Related Specifications

Designation	Title
CM-SP-eDOCSIS	eDOCSIS™ Specification
CM-SP-DRFI	Downstream Radio Frequency Interface Specification
CM-SP-DTI	DOCSIS Timing Interface Specification
CM-SP-DEPI	Downstream External PHY Interface Specification
CM-SP-DSG	DOCSIS Set-Top Gateway Interface Specification
CM-SP-M-OSSI	M-CMTS Operations Support System Interface Specification
CM-SP-L2VPN	Layer 2 Virtual Private Networks Specification
CM-SP-TEI	TDM Emulation Interface Specification

1.3 Requirements

Throughout this document, the words that are used to define the significance of particular requirements are capitalized. These words are:

"MUST"	This word means that the item is an absolute requirement of this specification.
"MUST NOT"	This phrase means that the item is an absolute prohibition of this specification.
"SHOULD"	This word means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood, and the case carefully weighed before choosing a different course.
"SHOULD NOT"	This phrase means that there may exist valid reasons in particular circumstances when the listed behavior is acceptable or even useful, but the full implications should be understood, and the case carefully weighed before implementing any behavior described with this label.
"MAY"	This word means that this item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

This document defines many features and parameters, and a valid range for each parameter is usually specified. Equipment (CM) requirements are always explicitly stated. Equipment needs to comply with all mandatory (MUST and MUST NOT) requirements to be considered compliant with this specification. Support of non-mandatory features and parameter values is optional.

1.4 Conventions

In this specification the following convention applies any time a bit field is displayed in a figure. The bit field should be interpreted by reading the figure from left to right, then from top to bottom, with the MSB being the first bit so read and the LSB being the last bit so read.

```
SNMP MIB syntax is represented by this code sample font.
```

Note: Notices and/or Warnings are identified by this style font and label.

1.5 Organization of Document

Section 1 provides an overview of the DOCSIS 4.0 series of specifications including the DOCSIS reference architecture and statement of compatibility.

Section 2 includes a list of normative and informative references used within this specification.

Section 3 defines the terms used throughout this specification.

Section 4 defines the acronyms used throughout this specification.

Section 5 provides a technical overview and lists the DOCSIS 4.0 key features for the functional areas of this specification.

Section 6 defines requirements for the OSSI management protocols.

Section 7 defines the requirements for the OSSI management objects including SNMP MIBs.

Section 8 defines the FCAPS OSSI requirements for the PHY, MAC, and Network Layers.

Section 9 defines the OSSI requirements for the Cable Modem to CPE Interface (CMCI).

Section 10 defines the OSSI requirements for the Cable Modem device including LED operations.

1.5.1 Annexes (Normative)

Annex A includes a detailed list of MIB object requirements for the CM.

Annex B defines protocol filtering requirements.

Annex C includes a detailed list of DOCSIS events and the associated formats.

Annex D defines the information model for the DOCSIS 4.0 Proactive Network Maintenance feature.

Annex E defines the DOCSIS 4.0 data type definitions.

Annex F defines the information model for the CM status and interface requirements.

Annex G defines the information model for the CM MULPI requirements.

1.5.2 Appendices (Informative)

Appendix I identifies spectrum analysis use cases.

Appendix II includes acknowledgements and contains a list of contributors.

2 REFERENCES

2.1 Normative References

In order to claim compliance with this specification, it is necessary to conform to the following standards and other works as indicated, in addition to the other requirements of this specification. Notwithstanding, intellectual property rights may be required to use or implement such normative references.

[CANN]	CableLabs' Assigned Names and Numbers, CL-SP-CANN-I22-230308, March 8, 2023, Cable Television Laboratories, Inc.
[CCAP-OSSIv3.1]	DOCSIS Converged Cable Access Platform Operations Support System Interface Specification, CM-SP-CCAP-OSSIv3.1-I27-231012, October 12, 2023, Cable Television Laboratories, Inc.
[CCAP-OSSIv4.0]	DOCSIS Converged Cable Access Platform Operations Support System Interface Specification, CM-SP-CCAP-OSSIv4.0-I10-231012, October 12, 2023, Cable Television Laboratories, Inc.
[CMCIv3.0]	DOCSIS Cable Modem to Customer Premise Equipment Interface Specification, CM-SP-CMCIv3.0-I03-170510, May 10, 2017, Cable Television Laboratories, Inc.
[DOCS-BPI2EXT-MIB]	CableLabs DOCSIS DOCS-BPI2EXT-MIB SNMP MIB Module, DOCS-BPI2EXT-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DOCS-IFEXT2-MIB]	CableLabs DOCSIS DOCS-IFEXT2-MIB SNMP MIB Module, DOCS-IFEXT2-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DOCS-IF3-MIB]	CableLabs DOCSIS DOCS-IF3-MIB SNMP MIB Module, DOCS-IF3-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DOCS-IF31-MIB]	CableLabs DOCSIS DOCS-IF31-MIB SNMP MIB Module, DOCS-IF31-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DOCS-PNM-MIB]	CableLabs DOCSIS DOCS-PNM-MIB SNMP MIB Module, DOCS-PNM-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DOCS-QOS3-MIB]	CableLabs DOCSIS DOCS-QOS3-MIB SNMP MIB Module, DOCS-QOS3-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DOCS-SEC-MIB]	CableLabs DOCSIS DOCS-SEC-MIB SNMP MIB Module, DOCS-SEC-MIB, http://www.cablelabs.com/MIBs/DOCSIS/.
[DSG]	DOCSIS Set-Top Gateway (DSG) Interface Specification, CM-SP-DSG-I25-170906, September 6, 2017, Cable Television Laboratories, Inc.
[IEEE802.1Q]	IEEE 802.1Q-2018, IEEE Standard for Local and Metropolitan Area Networks - Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks, July 2018.
[IPDR/SSDG]	IPDR Service Specification Design Guide, Version 3.8, TM Forum, October 2009.
[IPDR/XDR]	IPDR/XDR File Encoding Format, Version 3.5.1, TM Forum, October 2009.
[ISO8859-1]	ISO/IEC 8859-1:1998 Information technology—8-bit single-byte coded graphic character sets—Part 1: Latin alphabet No. 1, April 1998.
[M-OSSI]	DOCSIS M-CMTS Operations Support System Interface Specification, CM-SP-M-OSSI-I08-081209, December 9, 2008, Cable Television Laboratories, Inc.
[MULPIv3.1]	DOCSIS MAC and Upper Layer Protocols Interface Specification, CM-SP-MULPIv3.1-I25-230419, April 19, 2023, Cable Television Laboratories, Inc.
[MULPIv4.0]	DOCSIS MAC and Upper Layer Protocols Interface Specification, CM-SP-MULPIv4.0-I07-230503, May 3, 2023, Cable Television Laboratories, Inc.
[PHYv3.0]	DOCSIS Physical Layer Specification, CM-SP-PHYv3.0-C01-171207, December 7, 2017, Cable Television Laboratories, Inc.
[PHYv3.1]	DOCSIS Physical Layer Specification, CM-SP-PHYv3.1-I20-230419, April 19, 2023, Cable Television Laboratories, Inc.
[PHYv4.0]	DOCSIS Physical Layer Specification, CM-SP-PHYv4.0-I06-221019, October 19, 2022, Cable Television Laboratories, Inc.
[RFC 1157]	IETF RFC 1157, A Simple Network Management Protocol (SNMP), May 1990.
[RFC 1901]	IETF RFC 1901, Introduction to Community-based SNMPv2, January 1996.
[RFC 2348]	IETF RFC 2348, TFTP Blocksize Option, May 1998.
[RFC 2578]	IETF RFC 2578, Structure of Management Information Version 2 (SMIv2), April 1999.
[RFC 2580]	IETF RFC 2580, Conformance Statements for SMIv2, April 1999.

[RFC 2669] IETF RFC 2669, DOCSIS Cable Device MIB Cable Device Management Information Base for DOCSIS compliant Cable Modems and Cable Modem Termination Systems, August 1999.

[RFC 2786] IETF RFC 2786, Diffie-Hellman USM Key Management Information Base and Textual Convention, March 2000.

[RFC 2790] IETF RFC 2790, Host Resources MIB, March 2000.

[RFC 2863] IETF RFC 2863, The Interfaces Group MIB, June 2000.

[RFC 2933] IETF RFC 2933, Internet Group Management Protocol MIB, October 2000.

[RFC 3083] IETF RFC 3083, Baseline Privacy Interface Management Information Base for DOCSIS Compliant Cable Modems and Cable Modem Termination Systems, March 2001.

[RFC 3164] IETF RFC 3164, The BSD syslog Protocol, August 2001.

[RFC 3279] IETF RFC 3279, Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, April 2002.

[RFC 3410] IETF RFC 3410, Introduction and Applicability Statements for Internet-Standard Management Framework, December 2002.

[RFC 3411] IETF RFC 3411/STD0062, An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks, December 2002.

[RFC 3412] IETF RFC 3412, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), December 2002.

[RFC 3413] IETF RFC 3413/STD0062, Simple Network Management Protocol (SNMP) Applications, December 2002.

[RFC 3414] IETF RFC 3414/STD0062, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), December 2002.

[RFC 3415] IETF RFC 3415, View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP), December 2002.

[RFC 3416] IETF RFC 3416, Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP), December 2002.

[RFC 3417] IETF RFC 3417, Transport Mappings for the Simple Network Management Protocol (SNMP), December 2002.

[RFC 3418] IETF RFC 3418, Management Information Base (MIB) for the Simple Network Management Protocol (SNMP), December 2002.

[RFC 3419] IETF RFC 3419, Textual Conventions for Transport Addresses, December 2002.

[RFC 3433] IETF RFC 3433, K.C. Norseth, Entity Sensor Management Information Base, December 2002.

[RFC 3584] IETF RFC 3584, Coexistence between Version 1, Version 2, and Version 3 of the Internet-Standard and Network Management Framework, March 2000.

[RFC 3635] IETF RFC 3635, Definitions of Managed Objects for the Ethernet-like Interface Types, September 2003.

[RFC 3826] IETF RFC 3826, The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model, June 2004.

[RFC 3927] IETF RFC 3927, Dynamic Configuration of IPv4 Link-Local Addresses, May 2005.

[RFC 4022] IETF RFC 4022, Management Information Base for the Transmission Control Protocol (TCP), March 2005.

[RFC 4113] IETF RFC 4113, Management Information Base for the User Datagram Protocol (UDP), June 2005.

[RFC 4131] IETF RFC 4131, Management Information Base for Data Over Cable Service Interface Specification (DOCSIS) Cable Modems and Cable Modem Termination Systems for Baseline Privacy Plus, September 2005.

[RFC 4188] IETF RFC 4188, Definitions of Managed Objects for Bridges, September 2005.

[RFC 4253] IETF RFC 4253, The Secure Shell (SSH) Transport Layer Protocol, January 2006.

[RFC 4293] IETF RFC 4293, Management Information Base for the Internet Protocol (IP), April 2006.

[RFC 4546] IETF RFC 4546, Radio Frequency (RF) Interface Management Information Base for DOCSIS 2.0 Compliant RF Interfaces, June 2006.

[RFC 4639] IETF RFC 4639, Cable Device Management Information Base for Data-Over-Cable Service Interface Specification (DOCSIS) Compliant Cable Modems and Cable Modem Termination Systems, December 2006.

[RFC 5280] IETF RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008.

[RFC 6933] IETF RFC 6933, Entity MIB (Version 4), May 2013.

[SECv3.1] DOCSIS Security Specification, CM-SP-SECv3.1-I11-230419, April 19, 2023, Cable Television Laboratories, Inc.

[SECv4.0] DOCSIS Security Specification, CM-SP-SECv4.0-I06-230503, May 3, 2023, Cable Television Laboratories, Inc.

[USB] Universal Serial Bus Specification, Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips, Revision 2.0, April 27, 2000 (http://www.usb.org)

2.2 Informative References

This specification uses the following informative references.

[ISO11404] ISO/IEC 11404:1996 Information technology--Programming languages, their environments and system software interfaces--Language-independent datatypes, January 2002.

[ISO19501] ISO/IEC 19501:2005 Information technology -- Open Distributed Processing -- Unified Modeling Language (UML) Version 1.4.2.

[ITU-TX.692] ITU-T Recommendation X.692 (03/2002), Information technology - ASN.1 encoding rules: Specification of Encoding Control Notation (ECN).

[ITU-T M.3400] ITU-T Recommendation M.3400 (02/2000), TMN management functions.

[RFC 791] IETF RFC 791, Internet Protocol, September 1981.

[RFC 1213] IETF RFC 1213, Management Information Base for Network Management of TCP/IP-based internets: MIB-II, March 1991.

[RFC 1350] IETF RFC 1350, TFTP Protocol (Revision 2), July 1992.

[RFC 2460] IETF RFC2460, Internet Protocol, Version 6 (IPv6) Specification. S. Deering and R. Hinden, December 1998.

[RFC 2579] IETF RFC 2579, Textual Conventions for SMIv2, April 1999.

[RFC 2856] IETF RFC 2856, Textual Conventions for Additional High Capacity Data Types, June 2000.

[RFC 3168] IETF RFC3168, The Addition of Explicit Congestion Notification.

[RFC 3260] IETF RFC 3260, New Terminology and Clarifications for Diffserv, April 2002.

[RFC 3289] IETF RFC 3289, Management Information Base for the Differentiated Services Architecture, May 2002.

[RFC 4001] IETF RFC 4001, Textual Conventions for Internet Network Addresses, February 2005.

[RFC 4181] IETF RFC 4181, Guidelines for Authors and Reviewers of MIB Documents, September 2005.

[RFC 4291] IETF RFC 4291, Internet Protocol Version 6 (IPv6) Addressing Architecture, February 2006.

[SCTE RP] SCTE Measurement Recommended Practices for Cable Systems, Fourth Edition, March 2012, http://www.scte.org/ItemDetail?iProductCode=TS46

[Slope/Ripple] OSSIv3.1 Slope and Ripple Fit Calculation Example, http://www.cablelabs.com/specification/cable-modem-operations-support-system-interface-specification/

[UML Guidelines] UML Modeling Guidelines, CM-GL-OSS-UML-V01-180627, June 27, 2018, Cable Television Laboratories, Inc.

[X.509] ITU-T Recommendation X.509 (10/12): Information Technology - Open Systems Interconnection - The Directory: Public key and attribute certificate frameworks.

2.3 Reference Acquisition

CableLabs Specifications:

- Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027; Phone +1-303-661-9100, Fax +1-303-661-9199; http://www.cablelabs.com

SCTE•ISBE:

- Society of Cable Telecommunications Engineers Inc., 140 Philips Road, Exton, PA 19341; Phone: 610-363-6888 / 800-542-5040; Fax: 610-363-5898; http://www.scte.org/

IETF Specifications:

- Internet Engineering Task Force (IETF) Secretariat, 48377 Fremont Blvd., Suite 117, Fremont, California 94538, USA; Phone: +1-510-492-4080, Fax: +1-510-492-4001; http://www.ietf.org

ISO Specifications:

- International Organization for Standardization (ISO), 1, rue de Varembé, Case postale 56, CH-1211 Geneva 20, Switzerland; Phone +41 22 749 01 11, Fax +41 22 733 34 30; http://www.iso.org

ITU Recommendations:

- International Telecommunication Union, Place des Nations, CH-1211, Geneva 20, Switzerland; Phone +41-22-730-51-11; Fax +41-22-733-7256; http://www.itu.int

TM Forum:

- 240 Headquarters Plaza, East Tower, 10th Floor, Morristown, NJ 07960-6628; Phone: +1 973-944-5100, Fax: +1 973-944-5110; http://www.tmforum.org/DownloadCenter/7549/home.html#ipdr

3 TERMS AND DEFINITIONS

This specification uses the following terms.

Active Queue Management	AQM schemes attempt to maintain low queue occupancy (within Downstream and Upstream service flows) while supporting the ability to absorb a momentary traffic burst.
Allocation	A group of contiguous minislots in a MAP which constitute a single transmit opportunity.
Burst	A single continuous RF signal from the upstream transmitter, from transmitter on to transmitter off.
Cable Modem (CM)	A modulator-demodulator at subscriber locations intended for use in conveying data communications on a cable television system.
Cable Modem Termination System (CMTS)	Cable modem termination system, located at the cable television system headend or distribution hub, which provides complementary functionality to the cable modems to enable data connectivity to a wide-area network.
Cable Modem to CPE Interface (CMCI)	The interface, defined in [CMCIv3.0], between a CM and CPE.
Carrier-to-Noise plus Interference Ratio (CNIR)	The ratio of the expected commanded received signal power at the CMTS input to the noise plus interference in the channel.
Channel	The frequency spectrum occupied by a signal. Usually specified by center frequency and bandwidth parameters.
Classifier	A set of criteria used for packet matching according to TCP, UDP, IP, LLC, and/or 802.1P/Q packet fields. A classifier maps each packet to a Service Flow. A Downstream Classifier is used by the CMTS to assign packets to downstream service flows. An Upstream Classifier is used by the CM to assign packets to upstream service flows.
Customer	See End User.
Customer Premises Equipment (CPE)	Equipment at the end user's premises; may be provided by the end user or the service provider.
Data Model	A Data Model (as opposed to an Information Model) is defined at a lower level of abstraction, intended for implementations, and includes protocol-specific constructs. Since conceptual models can be implemented in different ways, multiple Data Models can be derived from a single Information Model. Data Models are technology specific. The Cable Modem has defined Data Models for SNMP as SNMP MIB modules.
Downstream (DS)	In cable television, the direction of transmission from the headend to the subscriber.
End User	A human being, organization, or telecommunications system that accesses the network in order to communicate via the services provided by the network.
FCAPS	A set of principles for managing networks and systems, wherein each letter represents one principle. F is for Fault, C is for Configuration, A is for Accounting, P is for Performance, S is for Security.
Fiber Node	A point of interface between a fiber trunk and the coaxial distribution.
Hybrid Fiber/Coax (HFC) System	A broadband bidirectional shared-media transmission system using fiber trunks between the headend and the fiber nodes, and coaxial distribution from the fiber nodes to the customer locations.
Inform	A confirmed SNMP message for asynchronous notification of events from an SNMP entity.
Information Model	An Information Model (as opposed to a Data Model) is an abstraction and only provides a high-level view of things of interest (i.e., information) to the business. It aids in understanding the scope and breadth of the business, rather than the depth. An Information Model is a way of representing and structuring information that has advantages over other common artifacts such as a glossary, descriptive document, database, or source code. A common Information Model will streamline the processes associated with information exchange, both within a business (e.g., Enterprise) and between the business and its external stakeholders.
International Organization for Standardization (ISO)	An international standards body, commonly known as the International Standards Organization.
Local Log	A volatile or non-volatile log stored within a network element.
Logical Upstream Channel	A MAC entity identified by a unique channel ID and for which bandwidth is allocated by an associated MAP message. A physical upstream channel may support multiple logical upstream channels. The associated UCD and MAP messages completely describe the logical channel.
Media Access Control (MAC) address	The "built-in" hardware address of a device connected to a shared medium.

MAC Domain	A subcomponent of the CMTS that provides data forwarding services to a set of downstream and upstream channels.
MAC Domain Downstream Service Group	The subset of a Downstream Service Group (DS-SG) which is confined to the Downstream Channels of a single MAC domain. An MD-DS-SG differs from a DS-SG only when multiple MAC domains are configured per CM-SG.
MAC Domain Upstream Service Group	The subset of an Upstream Service Group (US-SG) which is confined to the Upstream Channels of a single MAC Domain. An MD-US-SG differs from a US-SG only when multiple MAC domains are defined per CM-SG.
Micro-reflections	Echoes in the forward or reverse transmission path due to impedance mismatches between the physical plant components. Micro-reflections are distinguished from discrete echoes by having a time difference (between the main signal and the echo) on the order of 1 microsecond. Micro-reflections cause departures from ideal amplitude and phase characteristics for the transmission channel.
Minislot	A "minislot" is an integer multiple of 6.25-microsecond increments.
Network Management	The functions related to the management of data link layer and physical layer resources and their stations across the data network supported by the hybrid fiber/coax system.
Network Management System (NMS)	The hardware and software components used by the Network Provider to manage its networks as a whole. The Network Management System provides an end-to-end network view of the entire network enabling management of the network elements contained in the network.
Notification	Information emitted by a managed object relating to an event that has occurred within the managed object.
Occupied Bandwidth	1) Downstream - The sum of the bandwidth in all standard channel frequency allocations (e.g., 6 MHz spaced CEA channels) that are occupied by the OFDM channel. Even if one active subcarrier of an OFDM channel is placed in a given standard channel frequency allocation, that standard channel frequency allocation in its entirety is said to be occupied by the OFDM channel. 2) Upstream - a) For a single OFDMA channel, the sum of the bandwidth in all the subcarriers of that OFDMA channel which are not excluded. The upstream occupied bandwidth is calculated as the number of subcarriers which are not excluded, multiplied by the subcarrier spacing. b) For the transmit channel set, the sum of the occupied bandwidth of all OFDMA channels plus the bandwidth of the legacy channels (counted as 1.25 times the modulation rate for each legacy channel) in a cable modem's transmit channel set. The combined bandwidth of all the minislots in the channel is normally smaller than the upstream occupied bandwidth due to the existence of unused subcarriers. The bandwidth occupied by an OFDMA probe with a skip value of zero is equal to the upstream occupied bandwidth. [PHYv4.0].
Open Systems Interconnection (OSI)	A framework of ISO standards for communication between different systems made by different vendors, in which the communications process is organized into seven different categories that are placed in a layered sequence based on their relationship to the user. Each layer uses the layer immediately below it and provides a service to the layer above. Layers 7 through 4 deal with end-to-end communication between the message source and destination, and layers 3 through 1 deal with network functions.
Physical (PHY) Layer	Layer 1 in the Open System Interconnection (OSI) architecture; the layer that provides services to transmit bits or groups of bits over a transmission link between open systems and which entails electrical, mechanical and handshaking procedures.
PNM Server	One or more software application(s) for initiating PNM test and queries involving network elements, acting as a server from the perspective of other PNM and OSS applications, but acting as a client for network elements and measurement devices providing PNM and OSS results.
Pre-3.0 DOCSIS	Versions of CableLabs Data-Over-Cable-Service-Interface-Specifications (DOCSIS) prior to the DOCSIS 3.0 suite of specifications.
Primary Service Flow	All CMs have a Primary Upstream Service Flow and a Primary Downstream Service Flow. They ensure that the CM is always manageable, and they provide a default path for forwarded packets that are not classified to any other Service Flow.
Proactive Network Maintenance	The process and mechanism of measuring and assessing network conditions of the cable plant to determine error or fault conditions before becoming service impacting.
QoS Parameter Set	The set of Service Flow Encodings that describe the Quality of Service attributes of a Service Flow or a Service Class.
Service Class	A set of queuing and scheduling attributes that is named and that is configured at the CMTS. A Service Class is identified by a Service Class Name. A Service Class has an associated QoS Parameter Set.
Service Class Name	An ASCII string by which a Service Class may be referenced in modem configuration files and protocol exchanges.

Service Flow	A MAC-layer transport service which provides unidirectional transport of packets from the upper layer service entity to the RF and shapes, polices, and prioritizes traffic according to QoS traffic parameters defined for the Flow.
Service Flow Identifier (SFID)	A 32-bit identifier assigned to a service flow by the CMTS.
Service Identifier (SID)	A 14-bit identifier assigned by the CMTS to an Active or Admitted Upstream Service Flow.
Simple Network Management Protocol (SNMP)	A network management protocol of the IETF.
SNMP Agent	The term "agent" is used throughout this document to refer to 1) a SNMPv1/v2 agent or 2) a SNMPv3 entity [RFC 3411] which contains command responder and notification originator applications.
SNMP Manager	The term "manager" is used throughout this document to refer to 1) a SNMPv1/v2 manager or 2) a SNMPv3 entity [RFC 3411] which contains command generator and/or notification receiver applications.
Subscriber	See End User.
Syslog	A protocol that provides the transport of event notifications messages across IP networks.
Trap	An unconfirmed SNMP message for asynchronous notification of events from an SNMP entity.
Upstream (US)	The direction from the subscriber location toward the headend.

4 ABBREVIATIONS AND ACRONYMS

This specification uses the following abbreviations.

AAA	Network Authentication, Authorization, and Accounting
ACK	Acknowledge
ACM	Access Control Model
ADC	Analog-to-Digital Converter
AQM	Active Queue Management
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation 1
ATDMA	Advanced Time Division Multiple Access
BOOTR	Boot ROM
BPI	Baseline Privacy Interface
BPI+	Baseline Privacy Interface Plus
BPKM	Baseline Privacy Key Management
BSS	Business Support System
CA	Certificate Authority
CATV	Community Access Television, Cable Television
CCAP	Converged Cable Access Platform
CDC	Communications Device Class
CDS	Credential Data Structure
CLI	Command Line Interface
CM	Cable Modem
CMCI	Cable Modem to CPE Interface
CMIM	Cable Modem Interface Mask
CM-SG	Cable Modem Service Group
CMTS	Cable Modem Termination System
CNIR	Carrier-to-Noise plus Interference Ratio
CoS	Class of Service
CP	Cyclic Prefix
CPE	Customer Premises Equipment
CPU	Central Processing Unit
CRL	Certificate Revocation List
CSR	Customer Service Representative
CVC	Code Verification Certificate
dB	Decibel
DBC	Dynamic Bonding Change
DBG	Downstream Bonding Group
DCC	Dynamic Channel Change
DCID	Downstream Channel Identifier
DEPI	Downstream External Physical layer Interface
DFT	Discrete Fourier Transform
DH	Diffie-Hellman
DHCP	Dynamic Host Configuration Protocol
DLS	DOCSIS Light Sleep
DNS	Domain Name Service
DoS	Denial of Service
DS	Downstream

DSAP	Destination Service Access Point
DSCP	Differentiated Services Code Point
DSID	Downstream Service Identifier
DTD	Document Type Definition
EAE	Early Authentication and Encryption
ECC	Elliptic-curve Cryptography
ERMI	Edge Resource Manager Interface
eSAFE	Embedded Service/Application Functional Entity
EUI-64	64-bit Extended Unique Identifier
ECN	Explicit Congestion Notification
EM	Energy Management
FC	Frame Control
FCAPS	Fault, Configuration, Accounting, Performance, Security
FDX	Full Duplex or Full Duplex DOCSIS
FEC	Forward Error Correction
FFT	Fast Fourier Transform
FSM	Finite State Machine
HFC	Hybrid Fiber/Coax (HFC) System
HMAC	Keyed-Hash Message Authentication Code
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ID	Identifier
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IM	Information model
INIT	Initialize or Initialization
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPDR	Internet Protocol Detail Record
ISO	International Standards Organization
ITU	International Telecommunications Union
ITU-T	Telecommunication Standardization Sector of the International Telecommunication Union
LAN	Local Area Network
LED	Light Emitting Diode
LLC	Logical Link Control
LLD	Low Latency DOCSIS
LSB	Least Significant Bit
MAC	Media Access Control
MAP	Bandwidth Allocation Map
M-CMTS	Modular Cable Modem Termination System
MDD	MAC Domain Descriptor
MD-DS-SG	MAC Domain Downstream Service Group
MD-US-SG	MAC Domain Upstream Service Group
MER	Modulation Error Ratio
MIB	Management Information Base
MIC	Message Integrity Check
MP	Multipart

MSB	Most Significant Bit
MSO	Multiple Systems Operator
MTA	Multimedia Terminal Adapter
MTC	Multiple Transmit Channel
MULPI	MAC and Upper Layer Protocols Interface
NACO	Network Access Control Object
NE	Network Element
NMS	Network Management System
NSI	Network Side Interface
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OSI	Open Systems Interconnection
OSS	Operations Support System
OSSI	Operations Support System Interface
PC	Personal Computer
PCMM	PacketCable™ Multimedia
PDU	Protocol Data Unit
PGS	Proactive Grant Service
PHY	Physical Layer
PIE	Proportional Integral controller Enhanced
PKI	Public Key Infrastructure
PLC	PHY Link Channel
PNM	Proactive Network Maintenance
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase-Shift Keying
RBA	Resource Block Assignment
RCC	Receive Channel Configuration
RCP	Receive Channel Profile
RCP-ID	Receive Channel Profile Identifier
RCS	Receive Channel Set
REG	Registration
RFC	Request for Comments
RF	Radio Frequency
RFI	Radio Frequency Interface
RMS	Root Mean Square
RNG	Range or Ranging
ROM	Read Only Memory
RSA	Rivest, Shamir, Adleman (a public key cryptographic algorithm)
RxMER	Receive Modulation Error Ratio
SA	Security Association or Source Address
SAID	Security Association Identifier
SAMIS	Subscriber Accounting Management Interface Specification
SCCA	SSH Client Credential Authentication
S-CDMA	Synchronous Code Division Multiple Access
SCN	Service Class Name
SC-QAM	Single Carrier Quadrature Amplitude Modulation

SCTE	Society of Cable Telecommunications Engineers
SF	Service Flow
SFID	Service Flow Identifier
SG	Service Group
SID	Service Identifier
SLA	Service Level Agreement
SMIv1	Structure of Management Information Version 1
SMIv2	Structure of Management Information Version 2
SNAP	Sub-network Access Protocol
SNMP	Simple Network Management Protocol
SNMPv1	Version 1 of the Simple Network Management Protocol
SNMPv2	Version 2 of the Simple Network Management Protocol
SNMPv2c	Community-Based Simple Network Management Protocol, version 2
SNMPv3	Version 3 of the Simple Network Management Protocol
SNR	Signal to Noise Ratio
SSD	Secure Software Download
SSH	Secure Shell
STP	Spanning Tree Protocol
SW	Software
SYNC	Synchronize or Synchronization
TBD	To Be Determined (or To Be Deferred)
TCP	Transmission Control Protocol
TCS	Transmit Channel Set
TDMA	Time Division Multiple Access
TEK	Traffic Encryption Key
TFTP	Trivial File Transfer Protocol
TLV	Type/Length/Value
ToD	Time of Day
ToS	Type of Service
UBG	Upstream Bonding Group
UCC	Upstream Channel Change
UCD	Upstream Channel Descriptor
UCID	Upstream Channel Identifier
UDC	Upstream Drop Classifier
UDP	User Datagram Protocol
UML	Unified Modeling Language
URL	Uniform Resource Locator
US	Upstream
USB	Universal Serial Bus
USM	User-based Security Model
UTC	Coordinated Universal Time
VACM	View-based Access Control Model
VLAN	Virtual Local Area Network
XDR	External Data Representation
XML	Extensible Markup Language
XSD	XML Schema Definition

5 OVERVIEW

This section provides a brief description of the key management features in DOCSIS 4.0. These features are categorized according to the five conceptual categories of management developed as part of ITU Recommendation [ITU-T M.3400]. This set of management categories is referred to as the FCAPS model, represented by the individual management categories of Fault, Configuration, Accounting, Performance and Security.

In addition to the description of features, the rationale behind the introduction of information models is presented. Section 5.1 discusses the requirements introduced in DOCSIS 3.1, and Section 5.2 is a technical introduction to the detailed models in support of the user requirements.

5.1 DOCSIS 4.0 OSSI Key Features

Table 3 summarizes the requirements that support DOCSIS 4.0 features and the enhancements to existing management features. The table shows the management features along with the traditional Network Management Functional areas (Fault, Configuration, Accounting, Performance and Security) for the Network Elements (NE) Cable Modem (CM) and the corresponding OSI layer where those features operate.

Table 3 - Management Feature Requirements for DOCSIS 4.0

Features	Management Functional Area	OSI layer	Description
OFDM downstream signals and OFDMA upstream signals	Configuration	PHY	Provisioning physical downstream and upstream interfaces that support OFDM/OFDMA receivers according to their capabilities.
Plant Topology	Configuration	PHY, MAC (Data Link)	Provisioning flexible arrangements of US/DS channels for channel bonding configuration to reflect HFC plant topology.
Enhanced Diagnostics	Fault	PHY, MAC, Network	Expanded metrics for Proactive Network Maintenance (PNM).
Enhanced Performance Data Collection	Performance	PHY, MAC, Network	Collection of large statistical data sets for DOCSIS 4.0 feature sets.
Protective Network Maintenance and Enhanced Signal Quality Monitoring	Performance	PHY	To gather information on narrow band ingress and distortion affecting the quality of the RF signals.
Light Sleep Mode	Configuration	MAC	Energy efficiency mode for the Cable Modem to minimize power consumption.
Backup Primary Channels	Configuration	MAC	Retrieval of configuration status of backup downstream interfaces
Active Queue Management (AQM)	Configuration	MAC	Configuration of buffer management associated with service flows.
Low Latency	Configuration Performance	MAC	Service flow latency reporting

5.1.1 Fault Management Features

The DOCSIS 4.0 fault management requirements include:

- Extended lists of detailed events related to the new set of DOCSIS 4.0 features.
- Expanded metrics for Proactive Network Maintenance (PNM).

5.1.2 Configuration Management Features

The configuration of the DOCSIS protocols for CM/CMTS interactions for configuring features in support of PHY MAC/QoS and Security (BPI+) uses the CM configuration file and CMTS policies via MAC messages exchange. The reporting of configuration state information is done via SNMP MIB objects. This model provides a CM standard configuration with minimal operator intervention.

The DOCSIS 4.0 configuration requirements include:

- Updates to CM configuration parameters to support OFDM downstream interfaces, OFDMA upstream interfaces, light sleep mode and Active Queue Management (AQM).
- Retrieval of configuration status information for OFDM downstream interfaces, OFDMA upstream interfaces, light sleep mode, backup primary channels and Active Queue Management (AQM).
- Updates to CM QoS attributes to configure Service Flow Histogram calculations and reporting.

5.1.3 Performance Management Features

DOCSIS 4.0 requires an efficient mechanism for collecting large data sets as described above. The identified data sets are:

- Enhanced signal quality monitoring for granular plant status
- Statistics for dropped AQM packets
- Statistics for OFDM and OFMDA interface, subcarrier, profile and minislot counters
- Measurement Statistics for Proactive Network Maintenance (PNM)
- Latency statistics for histogram enabled service flows.

5.1.4 Security Management Features

Security Management includes both security of management information (e.g., SNMP access control) and management of network security related to authentication, authorization, and privacy of data plane communications. DOCSIS 3.1 defined a new certificate Public Key Infrastructure (PKI), with a set of management objects that strengthens the security of CM authentication and secure software download features.

The DOCSIS 4.0 security management features include:

- Secure Shell (SSH) Key Management, which provides a secure method to provision and manage the required set of credentials.
    - Transport Layer Security (TLS)-based Authentication (optional)
    - SNMP-based Authentication

Refer to [SECv4.0] for details on the SSH Key Management features.

5.1.5 Accounting Management Features

The DOCSIS 4.0 Accounting Management feature set is unchanged from DOCSIS 3.0.

5.2 Technical Overview

The technical overview presented in this section details functional areas of the FCAPS management model addressed by DOCSIS for managing the CM.

5.2.1 Architectural Overview

This section defines the functional areas of network management in terms of FCAPS (Fault, Configuration, Accounting, Performance and Security) as applied to the management of a CM within a DOCSIS network.

The requirements in the previous section were grouped both according to the management functional area and the relevant DOCSIS layer (using the OSI reference model) where they apply. This section provides an overview of the functions supported by each area. Even though specific functions are described for each area, there are interdependencies amongst all these functions to achieve the overall objective of efficient and proactive management of a CM in the DOCSIS network.

Fault management seeks to identify, isolate, correct and record system faults. Configuration management modifies system configuration variables and collects configuration information. Accounting management collects usage statistics for subscribers, sets usage quotas and bills users according to their use of the system. Performance management focuses on the collection of performance metrics, analysis of these metrics and the setting of thresholds and rate limits. Security management encompasses identification and authorization of users and equipment, provides audit logs and alerting functions, as well as providing vulnerability assessment.

Figure 4 illustrates the CM management architecture from the MSO back office interface perspective. The CM and CMTS reside within the Network Layer where services are provided to end Subscribers and various metrics are collected about network and service performance, among other things. Various management servers reside in the Network Management Layer within the MSO back office to provision, monitor and administer the Network Elements within the Network Layer (CM in this case). These management servers include, but are not limited to:

- The *SNMP Manager* performs SNMP configuration and queries against a CM's SNMP Agent.
- The *Configuration File Server* has the responsibility of transferring configuration files, via TFTP or optionally HTTP to the CM upon reinitialization.
- The *Firmware File Server* has the responsibility of transferring firmware images, according to the Secure Software Download mechanism, to a CM.
- The *Notification Receiver* receives autonomous SNMP notifications and Syslog messages from a CM.
- The *DHCP Server* has the responsibility of assigning a CM its IPv4 and/or IPv6 address as well as other DHCP parameters in order for the CM to obtain its configuration file and register on the network.
- The *Time Server* provides a CM with current Time of Day (ToD).
- The *IPDR Collector Servers* do not communicate directly with the CM. Rather, the CMTS collects various CM-related statistics and communicates this information to the IPDR Collector servers.
- The *TR-069 Server* does not communicate directly with the CM. Rather, if the CM is an eDOCSIS device and includes an eSAFE which supports the TR-069 protocol, the eDOCSIS device will communicate with the TR-069 server.

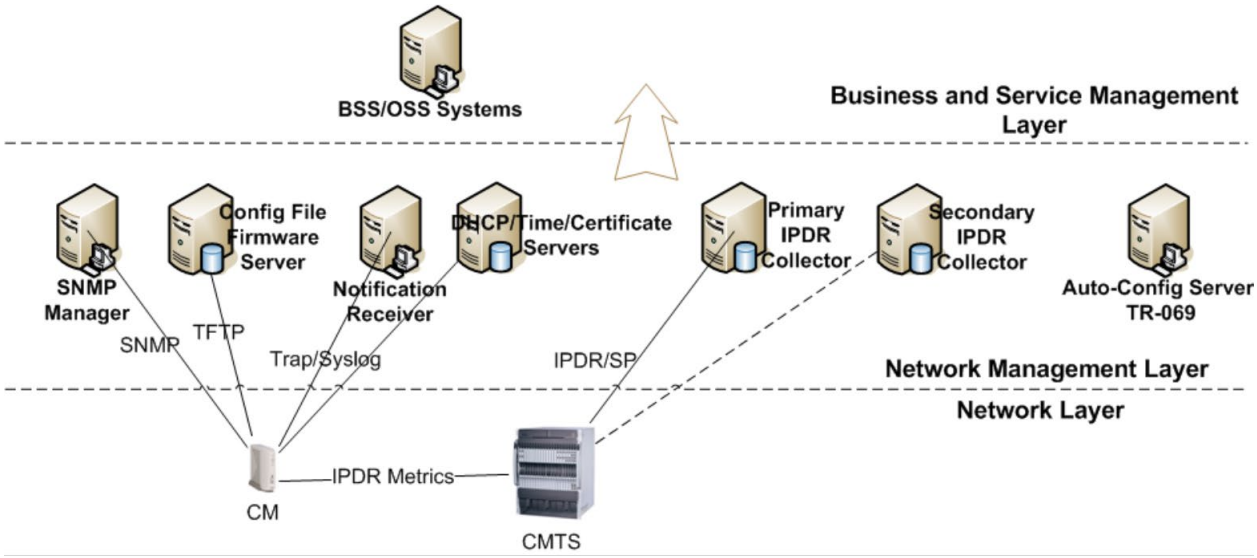


Figure 4 - CM Management Architecture

Finally, the Business and Service Management Layer is where higher level MSO business processes are implemented via BSS/OSS systems. These BSS/OSS systems utilize the data and information from the Network Management Layer which interrogated data from the Network Layer.

5.2.1.1 Fault Management

The goals of fault management are to provide failure detection, diagnosis, and perform or indicate necessary fault correction. Fault identification relies on the ability to monitor and detect problems, such as error-detection events. Fault resolution relies on the ability to diagnose and correct problems, such as executing a sequence of diagnostic test scripts, and correcting equipment or configuration faults. DOCSIS supports Event Reporting using Local Log, syslog, and SNMP notifications.

5.2.1.2 Configuration Management

Configuration management is concerned with adding, initializing, maintaining, and updating network elements. In a DOCSIS environment, network elements include CMs and CMTSs.

Configuration management is primarily concerned with network control via modifying operating parameters on network elements such as the CM and CMTS. Configuration parameters could include both physical resources (for example, an Ethernet interface) and logical objects (for example, QoS parameters for a given service flow).

While the network is in operation, configuration management is responsible for monitoring the configuration state and making changes in response to commands by a management system or some other network management function.

For example, a performance management function may detect that response time is degrading due to a high number of uncorrected frames, and may issue a configuration management change to modify the modulation type from 16-QAM to QPSK. A fault management function may detect and isolate a fault and may issue a configuration change to mitigate or correct that fault.

5.2.1.3 Accounting Management

Accounting management, in general, includes collection of usage data and permits billing the customer based on the subscriber's use of network resources. The CMTS is the network element that is responsible for providing the usage statistics to support billing. Billing is outside the scope of this specification.

5.2.1.4 Performance Management

Performance management functions include collecting statistics of parameters such as number of frames lost at the MAC layer and number of codeword errors at the PHY layer. These monitoring functions are used to determine the health of the network and whether the offered Quality of Service (QoS) to the subscriber is met. The quality of signal at the PHY layer is an indication of plant conditions.

The previous versions of DOCSIS OSSI specification defines SNMP polling as the collection mechanism for CM and CMTS statistics for performance management. SNMP polling of CMs is scalable and widely deployed with specialized engines that minimize the upstream bandwidth allocated to management during the polling intervals.

5.2.1.5 Security Management

Security management is concerned with both security of management information to protect the MSOs operations systems as well as managing the security information. The latter is used to authenticate and secure the traffic on the HFC. Security of the management interface is required to prevent end users from accessing and initiating configuration changes that may provide them with services for which they are not entitled or could result in the degradation or denial of services for other subscribers.

5.2.2 Management Protocols

As noted earlier in this section, the DOCSIS OSSI specification uses the Simple Network Management Protocol (SNMP) versions 1, 2c and 3 to define the management information for a CM DOCSIS network element in support of the functional areas mentioned in the previous section. SNMP is primarily a polling-based protocol where the management system retrieves data such as counter values and state information. There are events defined as a notification that are used to inform the management systems of fault conditions and security violations. The support for SNMP versions is continued in DOCSIS 4.0.

5.2.3 Information Models

The approach is based on an object-oriented modeling approach well known in the industry for capturing requirements and analyzing the data in a protocol independent representation. This approach defines requirements with use cases to describe the interactions between the operations support systems and the network element. The management information is represented in terms of objects along with their attributes and the interactions between these encapsulated objects (or also referred to as entities in some representations). The diagrams developed to capture these managed objects and their attributes and associations are UML Class Diagrams. The collection of UML Class Diagrams and Use Case Diagrams are referred to as the DOCSIS 4.0 Information Models. With the introduction of several new, complex features since DOCSIS 3.0 and the operator needs for a more proactive and efficient approach to management information, information modeling methodologies offer the ability to reuse the same definitions when new protocols are introduced in the future.

The managed objects are then represented in a protocol specific form referred to as a management data model. The management data models when using SNMP are described using the Structure of Management Information Version 2 (SMIv2) [RFC 2578] and the design of these models is determined by the capabilities of the protocol.

Refer to [UML Guidelines] for information on the modeling concepts used throughout this specification.

6 OSSI MANAGEMENT PROTOCOLS

6.1 SNMP Protocol

The SNMP protocol has been selected as the communication protocol for management of data-over-cable services.

The CM MUST implement the SNMPv3 protocol.

Although SNMPv3 offers certain security advantages over previous SNMP versions, many existing management systems do not fully support SNMPv3, necessitating support of the theoretically less secure but more ubiquitous SNMPv1 and SNMPv2c protocols.

The CM MUST implement the SNMPv1 and SNMPv2c protocols.

The IETF SNMP-related RFCs listed in Table 4 are supported by the CM.

Table 4 - IETF SNMP-related RFCs

[RFC 3410]	Introduction and Applicability Statements for Internet Standard Management Framework
[RFC 3411]	An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks
[RFC 3412]	Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)
[RFC 3413]	Simple Network Management Protocol (SNMP) Applications
[RFC 3414]	User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
[RFC 3415]	View-based Access Control Model (VACM) for the simple Network Management Protocol (SNMP)
[RFC 3416]	Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)
[RFC 3417]	Transport Mappings for the Simple Network Management Protocol (SNMP)
[RFC 3418]	Management Information Base for the Simple Network Management Protocol (SNMP)
[RFC 3419]	Textual Conventions for Transport Addresses
[RFC 3584]	Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework
[RFC 3826]	The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model
[RFC 1901]	Introduction to Community-based SNMPv2 (Informational)
[RFC 1157]	A Simple Network Management Protocol

For support of SMIv2, Table 5 lists the IETF SNMP-related RFCs which are supported by the CM.

Table 5 - SMIv2 IETF SNMP-related RFCs

[RFC 2578]	Structure of Management Information Version 2 (SMIv2)
[RFC 2579]	Textual Conventions for SMIv2
[RFC 2580]	Conformance Statements for SMIv2

For support of Diffie-Helman Key exchange for the User Based Security Model, Table 6 lists the IETF SNMP-related RFC which is supported by the CM.

Table 6 - Diffie-Helman IETF SNMP-related RFC

[RFC 2786]	Diffie-Helman USM Key Management Information Base and Textual Convention

6.1.1 Requirements for IPv6

Several transport domains were initially defined for SNMP (see [RFC 3417]). To support IPv6, [RFC 3419] adds a new set of transport domains not only for SNMP but for any application protocol.

The CM MUST support the recommendations of [RFC 3419] to support SNMP over IPv6.

7 OSSI MANAGEMENT OBJECTS

7.1 SNMP Management Information Bases (MIBS)

This section defines the minimum set of managed objects required to support the management of a CM.

The CM MAY augment the required MIBs with objects from other standard or vendor-specific MIBs where appropriate.

The DOCSIS 4.0 Cable Modem OSSI specification has priority over the IETF MIBs and all objects. Though deprecated or optional in the IETF MIB, the object can be required by this specification as mandatory.

The CM MUST implement the MIB requirements in accordance with this specification regardless of the value of an IETF MIB object's status (e.g., deprecated or optional).

If not required by this specification, deprecated objects are optional. If a CM implements a deprecated MIB object, the CM MUST implement the MIB object according to the MIB definition.

If a CM does not implement a deprecated MIB object, the following conditions MUST be met:

- The CM MUST NOT instantiate the deprecated MIB object.
- The CM MUST respond with the appropriate error/exception condition, such as noSuchObject for SNMPv2c, when an attempt to access the deprecated MIB object is made.

If not required by this specification, additional objects are optional. If a CM implements any additional MIB objects, the CM MUST implement the MIB object according to the MIB definition.

If a CM does not implement one or more additional MIB objects, the following conditions MUST be met:

- The CM MUST NOT instantiate the additional MIB object or objects.
- The CM MUST respond with the appropriate error/exception condition, such as noSuchObject for SNMPv2c when an attempt to access the non-existent additional MIB object is made, when the additional MIB object or objects are accessed.

If not required by this specification, obsolete objects are optional. If a CM implements an obsolete MIB object, the CM MUST implement the MIB object according to the MIB definition.

If a CM does not implement an obsolete MIB object, the following conditions MUST be met:

- The CM MUST NOT instantiate the obsolete MIB object.
- The CM MUST respond with the appropriate error/exception condition, such as noSuchObject for SNMPv2c, when an attempt to access the obsolete MIB object is made.

Objects which are not supported by this specification are not implemented by an agent.

- The CM MUST NOT instantiate MIB objects listed as not supported in Annex A.
- The CM MUST respond with the appropriate error/exception condition, such as noSuchObject for SNMPv2c, when an attempt to access a not supported MIB object is made.

Sections 7.1.1 and 7.1.2 include an overview of the MIB modules required for management of the facilities specified in the [MULPIv4.0] and [SECv4.0] specifications.

7.1.1 CableLabs MIB Modules

The following CableLabs MIB Modules are normatively required for CMs by this specification.

Table 7 - CableLabs MIB Modules

Reference	MIB Module
[DOCS-BPI2EXT-MIB]	DOCSIS Security MIB Module BPI+ Extension for DOCSIS 4.0 CM: DOCS-BPI2EXT-MIB
[DOCS-IFEXT2-MIB]	DOCSIS Interface Extension 2 MIB Module: DOCS-IFEXT2-MIB
[DOCS-IF3-MIB]	DOCSIS Interface 3 MIB Module: DOCS-IF3-MIB
[DOCS-IF31-MIB]	DOCSIS Interface 3.1 MIB Module, DOCS-IF31-MIB
[DOCS-PNM-MIB]	DOCSIS Proactive Network Maintenance MIB Module DOCS-PNM-MIB
[DOCS-QOS3-MIB]	DOCSIS Quality of Service 3 MIB Module: DOCS-QOS3-MIB
[DOCS-SEC-MIB]	DOCSIS Security MIB Module: DOCS-SEC-MIB

7.1.2 IETF RFC MIB Modules

Table 8 - IETF RFC MIB Modules

Reference	MIB Module
[RFC 2786]	Diffie-Helman USM Key MIB Module: SNMP-USM-DH-OBJECTS-MIB
[RFC 2790]	Host Resources MIB Module: HOST-RESOURCES-MIB
[RFC 2863]	Interfaces Group MIB Module: IF-MIB
[RFC 2933]	Internet Group Management Protocol MIB Module: IGMP-STD-MIB
[RFC 3410] [RFC 3411] [RFC 3412] [RFC 3413] [RFC 3414] [RFC 3415] [RFC 3584]	SNMPv3 MIB Modules: SNMP-FRAMEWORK-MIB, SNMP-MPD-MIB, SNMP-NOTIFICATION-MIB, SNMP-TARGET-MIB, SNMP-USER-BASED-SM-MIB, SNMP-VIEW-BASED-ACM-MIB, SNMP-COMMUNITY-MIB
[RFC 3418]	SNMPv2 MIB Module: SNMPv2-MIB
[RFC 3433]	Entity Sensor MIB Module: ENTITY-SENSOR-MIB
[RFC 3635]	Ethernet Interface MIB Module: EtherLike-MIB
[RFC 4022]	Transmission Control Protocol MIB Module: TCP-MIB
[RFC 4113]	User Datagram Protocol MIB Module: UDP-MIB
[RFC 4131]	DOCSIS Baseline Privacy Plus MIB Module: DOCS-IETF-BPI2-MIB
[RFC 4188]	Bridge MIB Module: BRIDGE-MIB
[RFC 4293]	Internet Protocol MIB Module: IP-MIB

Reference	MIB Module
[RFC 4546]	DOCSIS RF MIB Module: DOCS-IF-MIB
[RFC 4639]	DOCSIS Device MIB Module: DOCS-CABLE-DEVICE-MIB
[RFC 6933]	Entity MIB Module: ENTITY-MIB

7.1.3 Managed Objects Requirements

The following sections detail additional implementation requirements for the MIB modules listed.

The CM MUST implement the compliance and syntax of the MIB objects as specified in Annex A.

The CM MUST implement the MIB access requirements defined in Annex A as follows:

- MIB objects with Not-Accessible (N-Acc) access type are implemented with not-accessible access and are typically indexes in MIB tables.
- MIB objects with Read-Create (RC) access type are implemented with read-create access.
- MIB objects with Read-Write (RW) access type are implemented with read-write access.
- MIB objects with Read-Only (RO) access type are implemented with read-only access.
- MIB objects with Read-Create (RC) access type are implemented with read-create access.
- MIB objects with Read-Create (RC) or Read-Only (RO) access types are implemented with either read-create access or read-only access as described in the object.
- MIB objects with Read-Write (RW) or Read-Write (RO) access types are implemented with either read-write access or read-only access as described in the object.
- MIB objects with Accessible for SNMP Notification (Acc-FN) access type are implemented as SNMP Notifications or Traps.

The CM MUST support a minimum of 10 available SNMP table rows, unless otherwise specified by RFC or DOCSIS specification. The CM minimum number of available SNMP table rows SHOULD mean rows (per table) that are available to support device configuration. The CM used (default) SNMP table row entries MUST NOT apply to the minimum number of available SNMP table rows.

7.1.3.1 Requirements for CableLabs DOCSIS Interface Extension 2 MIB

The CM MUST implement DOCS-IFEXT2-MIB, as specified in [DOCS-IFEXT2-MIB].

7.1.3.2 Requirements for CableLabs DOCSIS Interface 3 MIB

The CM MUST implement the DOCS-IF3-MIB, as specified in [DOCS-IF3-MIB].

7.1.3.3 Requirements for CableLabs DOCSIS Interface 3.1 MIB

The CM MUST implement the DOCS-IF31-MIB, as specified in [DOCS-IF31-MIB].

7.1.3.4 Requirements for CableLabs DOCSIS Proactive Network Maintenance MIB

The CM MUST implement the DOCS-PNM-MIB, as specified in [DOCS-PNM-MIB].

7.1.3.5 Requirements for CableLabs DOCSIS Quality of Service 3 MIB

The CM MUST implement the DOCS-QOS3-MIB, as specified in [DOCS-QOS3-MIB].

7.1.3.6 Requirements for DOCSIS Device MIB (RFC 4639)

The CM MUST implement [RFC 4639].

Note: [RFC 4639] includes Compliance requirements for DIFFSERV-MIB [RFC 3289] to support IPv6 filtering as a replacement for the deprecated docsDevFilterIpTable. For backwards compatibility, this specification has requirements for docsDevFilterIpTable. IPv6 filtering requirements are specified in Annex A. This specification does not define requirements for [RFC 3289].

Additional requirements affecting [RFC 4639] are also found in Section 9.4.

7.1.3.7 Requirements for DOCSIS RF MIB (RFC 4546)

The CM MUST implement [RFC 4546]. However, much of [RFC 4546] is not applicable to OFDM/OFDMA channels. Thus, this section defines separate requirements for handling both SC-QAM and OFDM/OFDMA channels.

The CM MUST instantiate a row entry for all SC-QAM and OFDM channels in the docsIfDownChannelTable. The CM MUST return appropriate values for all columns of the docsIfDownChannelTable for SC-QAM channels as described in the MIB itself and further specified in this section.

OFDM channels are defined and configured differently than SC-QAM channels. Thus, the docsIfDownChannelTable cannot properly represent OFDM channels. However, it is useful for the NMS to have some representation of OFDM channels in the docsIfDownChannelTable as an indication that the channel exists and that more information can be found in other tables. Thus, rules are defined for OFDM channels to provide standard data via the docsIfDownChannelTable.

The CM MUST report the following values (Table 9 - docsIfDownChannelTable Requirements for OFDM Channels) for OFDM channel row entries in the docsIfDownChannelTable:

Table 9 - docsIfDownChannelTable Requirements for OFDM Channels

MIB Object	Value
docsIfDownChannelFrequency	0
docsIfDownChannelWidth	0
docsIfDownChannelModulation	other(2)
docsIfDownChannelInterleave	other(2)
docsIfDownChannelPower	0
docsIfDownChannelAnnex	other(2)
docsIfDownChannelStorageType	other(2)

For SC-QAM channels, the CM MUST implement the docsIfDownChannelPower MIB object with read-only access. For SC-QAM channels, the CM MUST report a power value for docsIfDownChannelPower within 3 dB of the actual received channel power when operated at nominal line-voltage, at normal room temperature (refer to [PHYv4.0]).

On SC-QAM channels, for any 1 dB change in input power, the CM MUST report a power change in the same direction that is not less than 0.6 dB and not more than 1.4 dB, as specified in [PHYv3.0].

Similarly, upstream OFDMA channels cannot be represented properly in the docsIfUpChannelTable. Thus, for OFDMA channels, the CM MUST report the following (Table 10 - docsIfUpChannelTable Requirements for OFDMA Channels) for OFDMA channel row entries in the docsIfUpChannelTable:

Table 10 - docsIfUpChannelTable Requirements for OFDMA Channels

MIB Object	Value
docsIfUpChannelFrequency	0
docsIfUpChannelWidth	0
docsIfUpChannelModulationProfile	0
docsIfUpChannelSlotSize	0

MIB Object	Value
docsIfUpChannelTxTimingOffset	0
docsIfUpChannelType	unknown(0)

Other values in the docsIfUpChannelTable for OFDMA channels are reported in an implementation-dependent manner. Operators are advised to not derive meaning from any other column in this table for rows whose columns match the values defined in Table 10.

The CM MUST report the docsIfSignalQualityTable for SC-QAM channels. The CM MUST NOT include row entries for OFDM channels in the docsIfSignalQualityTable.

The object docsIfDocsisBaseCapability was deprecated in DOCSIS 3.1.

The CM MUST extend the DOCS-IF-MIB textual convention DocsEqualizerData SYNTAX as follows: the OCTET STRING range restriction has been removed.

```
DocsEqualizerData ::= TEXTUAL-CONVENTION

         STATUS       current

         DESCRIPTION

             "This data type represents the equalizer data

              as measured at the receiver interface.

              The format of the equalizer follows the structure of the

              Transmit Equalization Adjust RNG-RSP TLV of DOCSIS RFI

              v2.0 :

              1 byte Main tap location 1..(n + m)

              1 byte Number of forward taps per symbol

              1 byte Number of forward taps: n

              1 byte Number of reverse taps: m


              Following are the equalizer coefficients:

              First, forward taps coefficients:

              2 bytes F1 (real),  2 bytes  F1 (imag)

              ...

              2 bytes Fn (real),  2 bytes  Fn (imag)


              Then, reverse taps coefficients:

              2 bytes D1 (real),  2 bytes  D1 (imag)

              ...


              2 bytes Dm (real),  2 bytes  Dm (imag)
```

```
            The equalizer coefficients are considered signed 16-bit
            integers in the range from -32768 (0x8000) to 32767
            (0x7FFF).

            DOCSIS specifications require up to a maximum of
            64 equalizer taps (n + m); therefore, this object size
             can get up 260 bytes (4 + 4x64).
            The minimum object size (other than zero) for a t-spaced
            tap with a minimum of 8 symbols will be 36 (4 + 4x8)."
        REFERENCE
           "Data-Over-Cable Service Interface Specifications: Radio
            Frequency Interface Specification SP-RFIv2.0-I10-051209,
            Figure 8-23."
        SYNTAX       OCTET STRING
```

The OCTET STRING range restriction has been removed.

7.1.3.8 Requirements for Interfaces Group MIB (RFC 2863)

The CM MUST implement the interface MIB [RFC 2863].

The ifType object associated with a DOCSIS interface can have the following enumerated values:

- CATV MAC interface: docsCableMacLayer (127)
- CATV downstream channel: docsCableDownstream (128)
- CATV M-CMTS downstream channel: docsCableMCmtsDownstream (229) (See [M-OSSI])
- CATV Downstream OFDM interface: docsOfdmDownstream (277)
- CATV upstream interface: docsCableUpStream (129)
- CATV logical upstream channel: docsCableUpstreamChannel (205)
- Upstream OFDMA interface: docsOfdmaUpstream (278)

7.1.3.8.1 Interface Organization and Numbering

Assigned interface numbers for DOCSIS-MAC and Ethernet (Ethernet-like interface) are used in the NMAccessTable to configure access policy at these interfaces. These configurations are generally encoded in the configuration file using TLV encoding.

The following statements define the CM interface-numbering scheme requirements:

> The CM MUST implement an instance of ifEntry for each configured DOCSIS-MAC interface, downstream interface, upstream interface, and for all of its LAN interfaces. If a DOCSIS-MAC interface consists of more than one upstream and downstream channel, the CM MUST populate the ifTable with a separate instance of ifEntry for each channel.
>
> The CM MAY fix LAN interfaces during the manufacturing process or determine these dynamically during the operation of the CM based on whether or not an interface has a CPE device attached to it.

If the CM has multiple CPE interfaces, but only one CPE interface that can be enabled at any given time, the CM MUST populate the ifTable to contain only the entry corresponding to the enabled or the default CPE interface.

The CM MUST populate the ifTable as specified in Table 74 - [RFC 2863] ifTable/ifXTable MIB-Object Details for Ethernet and USB Interfaces through Table 75 - [RFC 2863] ifTable/ifXTable MIB-Object Details for MAC and RF Interfaces of Section A.2. The CM MUST maintain entries in the ifTable for the downstream and upstream interfaces for which the CMTS has configured DS Receive Channels and US Transmit Channels, respectively, for this particular CM, and not for the total number of the CM receivers and transmitters the CM supports. CMTS configured Receive Channels and Transmit Channels for a CM are defined in [MULPIv4.0].

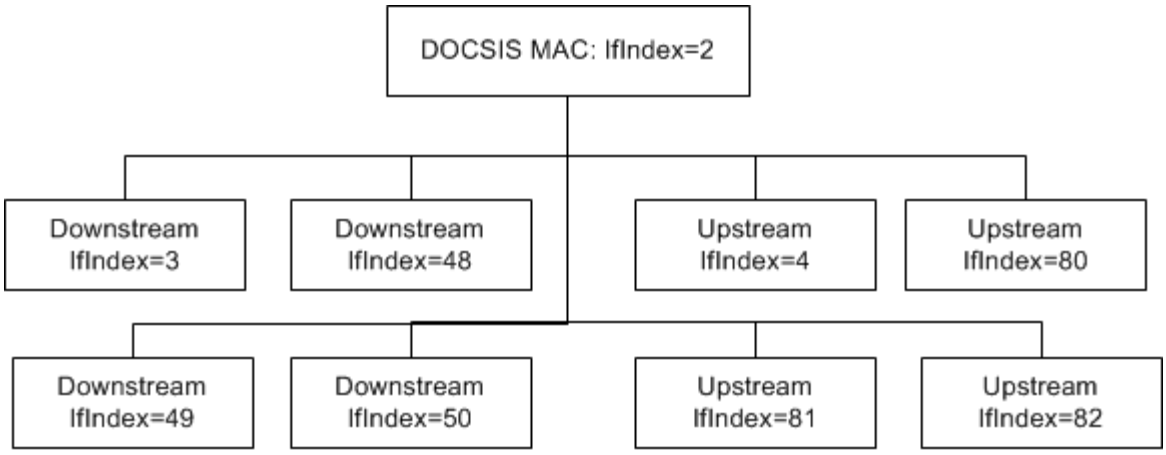
While the CM is registered, the CM SHOULD use a different ifIndex to allocate a new CMTS configured Receive Channel or Transmit Channel, and avoid the reuse of previously assigned IfIndexes that are not currently part of the CMTS configured Receive Channel Set (RCS) or Transmit Channel Set (TCS).

When a DS or US interface is configured as part of a RCS or TCS with a new channel id, the CM MUST update the ifCounterDiscontinuityTime and ifLastChange MIB variables.

The CM MUST populate ifStackTable with an entry for the DOCSIS-MAC interface and include the downstream and upstream interfaces are reported in the ifTable.

The CM MUST implement the MIB variable ifStackLastChange to report the value of sysUpTime where the ifStackTable change as a consequence of an addition or removal of a channel ID from a CM-SG as defined in [MULPIv4.0].

The following example illustrates a MAC interface with four downstream and four upstream interfaces for a CM.



Implementation of ifStackTable for this example:

ifStackHigherLayer	ifStackLowerLayer
0	2
2	3
2	4
2	48
2	49
2	50
2	80
2	81
2	82
3	0
4	0
48	0
49	0
50	0
80	0
81	0
82	0

Figure 5 - ifIndex Example for CM

The CM MUST number its interfaces as described in Table 11 - CM Interface Numbering.

Table 11 - CM Interface Numbering

Interface	Type
1	Primary CPE interface
2	DOCSIS-MAC interface
3	Primary downstream RF interface
4	One of the upstream RF interfaces
5 - 15	Additional CPE interfaces
16 - 31	eDOCSIS eSAFE interfaces
32 - 47	Additional CPE interfaces
48 - 79	Additional downstream RF interfaces
80 - 111	Additional upstream RF interfaces
112 - 143	Additional downstream RF interfaces

At any time, the CM MUST use ifIndex 3 for its primary downstream RF interface. The CM MUST use additional interface numbering sequentially starting from 48. If the interface numbers between 48-79 are exhausted, the CM MUST use 112-143 sequentially.

At any time, the CM MUST use ifIndex 4 for its first upstream RF interface. The CM MUST use additional interface numbering sequentially starting from 80.

For example, if the RCS is configured with channels on ifIndex 3 and 48 and the Dynamic Bonding Change DBC message demands ifIndex 3 be removed, the ifIndex 48 becomes ifIndex 3.

If the CM has more than one CPE interface, the vendor is required to define which of the CPE interfaces is the primary CPE interface. The CM is permitted to have its primary CPE interface fixed during the manufacturing process, or determine it dynamically during operation based on which interface has a CPE device attached to it. Regardless of the number of CPE interfaces the CM has, or how the primary CPE interface is determined, the CM will set the primary interface to interface number 1.

The CM MAY have additional CPE interfaces fixed during the manufacturing process or determined dynamically during operation based on which interface has a CPE device attached to it. Additional CPE interface ifIndexes are described in Table 11.

7.1.3.8.2 ifOperStatus Relationships

7.1.3.8.2.1 CmStatusValue and ifOperStatus Relationship

The CM MUST ensure that its CATV-MAC, downstream and upstream interfaces conform to the following relationships, as shown in Table 12 - CmStatusValue and ifOperStatus Relationship, of ifOperStatus and CmStatusValue (see Annex F) when ifAdminStatus value of those interfaces is 'up'.

Table 12 - CmStatusValue and ifOperStatus Relationship

IfOperStatus	CmStatusValue
'down'	'other', 'notReady'
'dormant'	'notSynchronized', 'phySynchronized', 'usParametersAcquired', 'rangingComplete', 'dhcpV4Complete', 'dhcpV6Complete', 'todEstablished', 'configFileDownloadComplete', 'startRegistration', 'bpiInit', 'accessDenied'
'up'	'registrationComplete', 'securityEstablished', 'operational'

7.1.3.8.2.2 USB state and ifOperStatus Relationships

If the CM supports USBs as CPE interfaces, the CM SHOULD report the value of the MIB object ifOperStatus as noted in Table 13 - USB State and ifOperStatus Relationship.

Table 13 - USB State and ifOperStatus Relationship

IfOperStatus	USB states and other conditions (see [USB])
'down'	'Attached', 'Powered', 'Default', and STALL operation
'dormant'	'Suspended', 'Address'
'up'	'Configured'

7.1.3.8.3 ifAdminStatus and Traffic

The CM MUST NOT accept or forward any traffic over an interface whose ifAdminStatus is 'down', (traffic includes data and MAC management traffic where applicable).

When the CM initializes, all interfaces start with ifAdminStatus in the up(1) state. As a result of explicit management action, ifAdminStatus is then changed to either the down(2) or testing(3) states (or remains in the up(1) state). As a result of either explicit management or configuration information saved via other non-SNMP method (i.e., CLI commands) retained by the managed system, ifAdminStatus is then changed to either the down(2) or testing(3) states (or remains in the up(1) state).

7.1.3.8.4 SNMP Notification Control Requirements

If a multi-layer interface model is present in the device, each sub-layer for which there is an entry in the ifTable can generate linkUp/Down traps. Since interface state changes would tend to propagate through the interface stack (from top to bottom, or bottom to top), it is likely that several traps would be generated for each linkUp/Down occurrence. The ifLinkUpDownTrapEnable object allows managers to control SNMP notification generation, and configure only the interface sub-layers of interest.

The CM MUST implement the MIB object ifLinkUpDownTrapEnable specified in [RFC 2863].

For linkUp/Down events on CM DOCSIS interfaces, the CM SHOULD generate an SNMP notification for the CM MAC interface and not for any sub-layers of the interface. Therefore, the CM MUST have its default setting of ifLinkUpDownTrapEnable for the CM MAC interface set to 'enabled'. The CM MUST have its default setting of ifLinkUpDownTrapEnable for the RF-Up interface(s) set to 'disabled'. The CM MUST have its default setting of ifLinkUpDownTrapEnable for the RF-Down interface(s) set to 'disabled'. The CM SHOULD have its default setting of ifLinkUpDownTrapEnable for interfaces 1 and 5 through 47 listed in Table 11 - CM Interface Numbering set to 'disabled'.

If the ifLinkUpDownTrapEnable for the CM MAC interface set to 'enabled', the CM MUST generate a linkUp SNMP notification [RFC 2863].

7.1.3.8.5 ifTable and IfXTable Counters

Application of the [RFC 2863] ifTable and ifXTable MIB counter objects are done on a per-interface basis for DOCSIS 3.0, DOCSIS 3.1, and DOCSIS 4.0 and are detailed in Table 74 and Table 75 of Annex A.2. These tables define specific SNMP Access and MIB requirements for each of the interface counters defined in [RFC 2863]. The CM MUST only count octets on the downstream and upstream interfaces. The CM MAY implement the packet counters from [RFC 2863], but when implemented on these interfaces, the counter object will return a value of zero. The CM ethernet and MAC interfaces count both packet and octet counters. Per the requirements in [RFC 2863] Counter Size section, a given interface may support only 32-bit or 64-bit (High Capacity), or both sets of counters based on interface speed.

The CM MUST implement the ifTable and ifXTable [RFC 2863] Counter32 and Counter64 MIB objects as defined for each interface in Table 76 - [RFC 2863] ifTable/ifXTable Counter32 and Counter64 MIB-Object Details for Ethernet and USB Interfaces and Table 77 - [RFC 2863] ifTable/ifXTable Counter32 and Counter64 MIB-Object Details for MAC and RF Interfaces of Section A.2.

The following table describes the rules for counting packets and octets on RF and MAC domain interfaces.

Table 14 - IF-MIB Counter Rules

MIB Counter Objects	MAC Domain Interfaces	Upstream/Downstream RF Interfaces
ifInOctets ifHCInOctets	The total number of data octets (data in transit, data targeted to the managed device) received on this interface from the RF interface and before application of protocol filters.	This includes MAC packets as well as data packets, and includes the length of the MAC header; this does not include any PHY overhead.
ifInUcastPkts ifHCInUcastPkts	The total number of Unicast data packets (data in transit, data targeted to the managed device) received on this interface from the RF interface before application of protocol filters.	Reports zero if implemented.
ifOutOctets ifHCOutOctets	The total number of data octets (data in transit, data generated by the managed device) transmitted on this interface to the RF interface after application of protocol filters.	This includes MAC packets as well as data packets, and includes the length of the MAC header; this does not include any PHY overhead.
ifOutUcastPkts ifHCOutUcastPkts	The total number of Unicast data packets (data in transit, data generated by the managed device) transmitted on this interface to the RF interface after application of protocol filters.	Reports zero if implemented.
ifInMulticastPkts ifHCInMulticastPkts	The total number of Multicast data packets (data in transit, data targeted to the managed device) received on this interface from the RF interface before application of protocol filters.	Reports zero if implemented.

MIB Counter Objects	MAC Domain Interfaces	Upstream/Downstream RF Interfaces
ifInBroadcastPkts ifHCInBroadcastPkts	The total number of Broadcast data packets (data in transit, data targeted to the managed device) received on this interface from the RF interface before application of protocol filters.	Reports zero if implemented.
ifOutMulticastPkts ifHCOutMulticastPkts	The total number of Multicast data packets (data in transit, data generated by the managed device) transmitted on this interface to the RF interface after application of protocol filters.	Reports zero if implemented.
ifOutBroadcastPkts ifHCOutBroadcastPkts	The total number of Broadcast data packets (data in transit, data generated by the managed device) transmitted on this interface to the RF interface after application of protocol filters.	Reports zero if implemented.

7.1.3.8.6 ifSpeed and ifHighSpeed

For SC-QAM downstream channels, the ifSpeed is the symbol rate multiplied by the number of bits per symbol. For SC-QAM upstream channels, the ifSpeed is the raw bandwidth in bits per second of this interface, regarding the highest speed modulation profile that is defined. This is the symbol rate multiplied with the number of bits per symbol for this modulation profile.

For OFDM downstream channels, the CM MUST calculate the ifSpeed per the following algorithm:

```
numCountedSubcarriers = 0;
totalBitLoading = 0;


for (i = 0; i < numActiveSubcarriers; ++i)

{
if subcarrier is not PLC and not continuous pilot then

{
totalBitLoading += modulationOrder(i); //in bits per symbol
numCountedSubcarriers += 1;

}

}


averageBitLoading = (totalBitLoading / numCountedSubcarriers);

ifSpeed = numCountedSubcarriers * averageBitLoading * subcarrierSpacingHz *
          [numFftSamples/(numFftSamples + cyclicPrefixSamples)] * (127/128)
```

The number of symbols in a minislot for a given OFDMA channel is a factor of the number of symbols in a frame and the number of subcarriers per minislot. The minislot capacity depends on the minislot bit loading and pilot pattern, which are variable per minislot based on IUC, the minislot location in the frame and the burst profile being used. Another factor is whether a minislot is classified as a body minislot or as an edge minislot.

For the purpose of calculating ifSpeed, the CM uses an OFDMA Data IUC with the highest capacity assuming that all minislots are body minislots. The minislot capacity is calculated by multiplying the number of data symbols in a minislot by modulation order (in bits per symbol) and adding to the number of complementary pilot symbols in a minislot multiplied by the complementary data pilot modulation order. The upstream channel capacity is calculated by adding the capacity of all minislots in a frame and multiplying that number by the frame rate.

For upstream OFDMA channels, the CM MUST calculate the ifSpeed per the following algorithm:

```
frameCapacity = 0;
```

```
for (i = 0; i < numMinislotsPerFrame; ++i)

{

minislotCapacity(i) = numDataSymbols(i) * modulationOrder(i) +
                              numComplementaryPilotSymbols(i) *
compPilotSymbolModulationOrder(i);
       frameCapacity += minislotCapacity(i);

}

numFramesPerSecond = 1/((numSymbolsPerOfdmaFrameK * fftDuration)*(idftSize/(idftSize +
cyclicPrefix))

ifSpeed = numFramesPerSecond * frameCapacity;
```

7.1.3.8.7 ifDescr

7.1.3.8.7.1 IfDescr for USB Interfaces

If the CM supports USB as CPE interfaces, the CM MUST report the value of the MIB object ifDescr for these interfaces as follows: USB <dbcUSB> CDC Ethernet; <any text>

<dbcUSB> corresponds to the USB version in the format JJ.M.N (JJ - major version number, M - minor version number, N - sub-minor version number). See Standard USB Descriptor Definitions from [USB] specification.

For example, if the dbcUSB field in the USB descriptor is 0x0213, <dbcUSB> is presented in ifDescrMib object as "2.1.3" and a value of 0x2000 in the dbcUSB field of the USB Descriptor is represented as "2.0" in ifDescr MIB object, in both cases without double quotes.

<Any text> indicates a vendor-specific text.

A complete example of ifDescr for an USB device is as follows (Assume dbcUBC 0x2000):

USB 2.0 CDC Ethernet; <any text>

7.1.3.9 Requirements for Ethernet Interface MIB (RFC 3635)

The CM MUST implement [RFC 3635] if Ethernet interfaces are present.

7.1.3.10 Requirements for Bridge MIB (RFC 4188)

The CM MUST implement the Bridge MIB [RFC 4188] to support the forwarding requirements defined in [MULPIv4.0].

The CM MUST implement a managed object (see docsDevSTPControl in [RFC 4639]) that controls the spanning tree protocol (STP) policy defined in [IEEE802.1Q] and in accordance with [MULPIv4.0] requirements.

If STP is enabled for the CM, then the CM implements the dot1dStp scalar group [RFC 4188] and optionally the dot1dStpPortTable [RFC 4188] as specified in Annex A.

7.1.3.11 Requirements for Internet Protocol MIB (RFC 4293)

The CM requirements for [RFC 4293] are defined in the following sections.

7.1.3.11.1 The IP Group

The CM MUST implement the ipv4GeneralGroup.

The CM MUST implement the ipv6GeneralGroup2.

The CM MUST implement the ipv4InterfaceTable.

The CM MUST populate the ipv4InterfaceTable with each Ethernet interface with an assigned IPv4 address. The CM MAY record other interfaces in the ipv4InterfaceTable which have assigned IPv4 addresses.

The CM MUST populate the ipv6InterfaceTable with each Ethernet interface with an assigned IPv6 address. The CM MAY record other interfaces in the ipv6InterfaceTable which have assigned IPv6 addresses.

The CM MAY implement the ipSystemStatsTable.

The CM MAY implement the ipIfStatsTable.

The CM MAY implement the ipAddressPrefixTable.

The CM MAY implement the ipAddressTable.

The CM MAY implement the ipNetToPhysicalTable.

The CM MAY implement the ipDefaultRouterTable.

The CM MUST NOT implement the ipv6RouterAdvertTable.

7.1.3.11.2 The ICMP Group

The CM MUST implement the icmpStatsTable.

The CM MUST implement the icmpMsgStatsTable.

7.1.3.12 Requirements for User Datagram Protocol MIB (RFC 4113)

The CM MAY implement the UDP-MIB in [RFC 4113].

7.1.3.13 Requirements for Transmission Control Protocol (TCP) MIB (RFC 4022)

7.1.3.13.1 The TCP Group

The CM MAY implement the TCP group in [RFC 4022].

7.1.3.14 Requirements for SNMPv2 MIB (RFC 3418)

7.1.3.14.1 The System Group

The CM MUST implement the System Group of [RFC 3418].

See Section 8.2.1 for sysDescr requirements details.

7.1.3.14.2 The SNMP Group

This group provides SNMP protocol statistics and protocol errors counters.

The CM MUST implement the SNMP Group from [RFC 3418].

7.1.3.15 Requirements for Internet Group Management Protocol MIB (RFC 2933)

DOCSIS 3.1 and DOCSIS 4.0 CMs are not required to implement [RFC 2933].

7.1.3.16 Requirements for DOCSIS Baseline Privacy Plus MIB (RFC 4131)

The CM MUST implement [RFC 4131].

The following MIB objects from [RFC 4131] are used to support legacy PKI CM certificate functions defined in the DOCSIS 3.0 security specification:

docsBpi2CmDeviceCmCert

docsBpi2CmDeviceManufCert

The CM MUST extend the DOCS-IETF-BPI2-MIB docsBpi2CmAuthRejectErrorCode object SYNTAX as follows.

```
docsBpi2CmAuthRejectErrorCode OBJECT-TYPE
```

```
SYNTAX          INTEGER {
                        none(1),
                        unknown(2),
                        unauthorizedCm(3),
                        unauthorizedSaid(4),
                        permanentAuthorizationFailure(8),
                        timeOfDayNotAcquired(11),
                        eaeDisabled(12),
                        unsupportedBpiVer(13)
                        }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
     "The value of this object is the enumerated
description of the Error-Code in the most recent
Authorization Reject message received by the CM.  This has
the value unknown(2) if the last Error-Code value was 0 and
none(1) if no Authorization Reject message has been received
since reboot."
REFERENCE
     "DOCSIS Baseline Privacy Plus Interface Specification
Sections 4.2.1.3 and 4.2.2.15."
::= { docsBpi2CmBaseEntry 22 }
```

The following enumerations have been added:

- eaeDisabled(12)
- unsupportedBpiVer(13)

7.1.3.17 Requirements for DOCSIS Baseline Privacy Plus Extension MIB [DOCS-BPI2EXT-MIB]

To support the new PKI defined in [SECv4.0], the CM MUST implement the DOCS-BPI2EXT-MIB defined in [DOCS-BPI2EXT-MIB], which is an extension to [RFC 4131].

The following MIB objects are defined in the [DOCS-BPI2EXT-MIB] module that supports PKI CM certificate functions defined in [SECv4.0]:

docsBpi2Ext31CmDeviceCmCert

docsBpi2Ext31CmDeviceManufCert

docsBpi2Ext31CodeUpdateCvcChain

docsBpi2Ext31CodeMfgOrgName

docsBpi2Ext31CodeMfgCodeAccessStart

docsBpi2Ext31CodeMfgCvcAccessStart

docsBpi2Ext31CodeCoSignerOrgName

docsBpi2Ext31CodeCoSignerCodeAccessStart

docsBpi2Ext31CodeCoSignerCvcAccessStart

7.1.3.18 Requirements for DOCSIS Security MIB [DOCS-SEC-MIB]

To support the SSH functionalities defined in [SECv4.0], the CM MUST implement the DOCS-SEC-MIB defined in [DOCS-SEC-MIB].

7.1.3.19 Requirements for Diffie-Helman USM Key MIB (RFC 2786)

The CM MUST implement [RFC 2786].

7.1.3.20 Requirements for DOCSIS Baseline Privacy MIB (RFC 3083)

A DOCSIS 4.0 CM is not required to implement [RFC 3083].

7.1.3.21 Requirements for SNMPv3 MIB Modules

The CM MUST implement the MIBs defined in [RFC 3411] through [RFC 3415] and [RFC 3584].

The CM MUST support the default value of 'volatile' for any SNMPv3 object with a StorageType syntax. This overrides the default value specified in [RFC 3411] through [RFC 3415] and [RFC 3584]. The CM MUST only accept the value of 'volatile' for any SNMPv3 object with a StorageType syntax. An attempted set to a value of 'other', 'nonVolatile', 'permanent', or 'readOnly' will result in an "inconsistentValue" error. Values other than the valid range (1-5) would result in a "wrongValue" error.

The CM SHOULD support a minimum of 30 available rows in the vacmViewTreeFamilyTable object.

7.1.3.22 Requirements for Entity MIB (RFC 6933)

The CM MAY implement the ENTITY-MIB [RFC 6933].

If the CM implements the ENTITY-SENSOR-MIB [RFC 3433], the CM is required to implement the entPhysicalTable with entries corresponding to any sensors managed in the ENTITY-SENSOR-MIB (e.g., temperature sensors). For sensor entries in the entPhysicalTable, the CM reports a value of 'sensor' for entPhysicalClass.

7.1.3.23 Requirements for Entity Sensor MIB (RFC 3433)

The CM MAY implement the Entity Sensor MIB [RFC 3433].

The CM MAY implement the entPhySensorTable for instances which exist in the entPhysicalTable of the ENTITY-MIB [RFC 6933] with an entPhysicalClass of 'sensor'. It is recommended that for temperature sensors, the CM report a value for entPhySensorType of 'celsius'.

7.1.3.24 Requirements for Host Resources MIB (RFC 2790)

The CM MAY implement the HOST-RESOURCES-MIB [RFC 2790].

8 OSSI FOR PHY, MAC AND NETWORK LAYERS

8.1 Fault Management

This section defines requirements for remote monitoring/detection, diagnosis, reporting, and correction of problems. Refer also to Section 7, for requirements for managed objects supporting CM fault management.

8.1.1 SNMP Usage

In the DOCSIS environment, SNMP is used to achieve the goals of fault management: remote detection, diagnosis, reporting, and correction of CM network faults. Therefore, the CM MUST support SNMP management traffic across the CATV MAC interfaces as long as the CM has ranged and registered.

The CM SNMP access might be restricted by configuration parameters to support the operator's policy goals. Cable operators' CM installation personnel can use SNMP queries from a station on the CMCI side to perform on-site CM and diagnostics and fault classification (note that this may require temporary provisioning of the CM from a local DHCP server). Further, CMCI side subscriber applications, using SNMP queries, can diagnose simple post-installation problems, avoiding visits from service personnel and minimizing help desk telephone queries.

The CM sends SNMP notifications to one or more NMSs (subject to operator-imposed policy). CM requirements for SNMP notifications are detailed in Section 8.1.2. The CM sends events to a syslog server. CM requirements for syslog events are detailed in Section 8.1.2.

8.1.2 Event Notification

A CM is required to generate asynchronous events that indicate malfunction situations and notify about important events. The methods for reporting events are defined below:

1. Stored in Local Log (docsDevEventTable [RFC 4639]).
2. Reported to SNMP entities as an SNMP notification.
3. Sent as a message to a syslog server.

This specification defines the support of DOCSIS specific events (see Annex C) and IETF events. The former are normally in the form of SNMP notifications. The delivery of IETF Notifications to local log and syslog server is optional.

Event Notifications are enabled and disabled by configuration. IETF SNMP notifications normally define specific controls to enable and disable notifications. For example, see Section 7.1.3.8.4 for requirements on ifLinkUpDownTrapEnable. DOCSIS specific events can be reported to local log and as syslog message and/or SNMNP notification as defined in docsDevEvControlTable [RFC 4639], Section 8.1.2.2, and Annex F, Section F.2.3.6. A CM supports event notification functions including local event logging, syslog (limiting/throttling) and SNMP notification (limiting/throttling), as specified in [RFC 4639] and this specification. A CM operating in SNMP v1/v2c NmAccess mode is required to support SNMP trap control as specified in [RFC 4639] and this specification. A CM operating in SNMP Coexistence mode is required to support SNMP notification functions, as specified in [RFC 3416] and [RFC 3413] and this specification.

8.1.2.1 Format of Events

Annex C lists all DOCSIS events.

The following sections explain in detail how to report these events by any of the three mechanisms (local event logging, SNMP notification and syslog).

8.1.2.1.1 Local Event Logging

A CM MUST maintain Local Log events, defined in Annex C, in both local-volatile storage and local non-volatile storage. A CM MAY retain in local non-volatile storage events designated for local volatile storage. A CM MAY retain in local volatile storage events designated for local non-volatile storage.

A CM MUST implement its Local Log as a cyclic buffer with a minimum of ten entries. The CM Local Log non-volatile storage events MUST persist across reboots. The CM MUST provide access to the Local Log events through the docsDevEventTable [RFC 4639].

Section 8.1.2.1.3 describes rules to generate unique EventIds from the error code.

The [RFC 4639] docsDevEvIndex object provides relative ordering of events in the log. The creation of local-volatile and local non-volatile logs necessitates a method for synchronizing docsDevEvIndex values between the two Local Logs after reboot. The CM MUST adhere to the rules listed below for creating local volatile and local non-volatile logs following a re-boot.

The CM MUST clear both the local volatile and local non-volatile event logs when an event log reset is initiated through an SNMP SET of the docsDevEvControl object [RFC 4639].

8.1.2.1.2 SNMP Notifications

A CM MUST implement the generic SNMP notifications according to Annex A.

When any event causes a generic SNMP notification occurrence in the CM, the CM MUST send notifications if throttling/limiting mechanisms defined in [RFC 4639] and other limitations [RFC 3413] do not restrict notification sending.

A CM MUST implement SNMP notifications defined in [DOCS-IF3-MIB].

A CM operating in SNMP v1/v2c NmAccess mode MUST support SNMPv1 and SNMPv2c Traps as defined in [RFC 3416].

A CM operating in SNMP Coexistence mode MUST support SNMP notification type 'trap' and 'inform' as defined in [RFC 3416] and [RFC 3413].

The CM MUST send notifications for any event, if docsDevEvControl object [RFC 4639], throttling/limiting mechanism [RFC 4639] and [RFC 3413] limitations applied later do not restrict notification sending.

The CM MUST NOT report via SNMP notifications vendor-specific events that are not described in instructions submitted with certification testing application documentation.

8.1.2.1.3 Syslog Message Format

When the CM sends a syslog message for a DOCSIS-defined event, the CM MUST send it in the following format:
<level>CABLEMODEM[vendor]: <eventId> text vendor-specific-text

Where:

- *level* is an ASCII representation of the event priority, enclosed in angle brackets, which is constructed as an OR of the default Facility (128) and event priority (0-7). The resulting level ranges between 128 and 135.
- *TIMESTAMP* and *HOSTNAME* follow the format of [RFC 3164]. The single space after *TIMESTAMP* is part of the *TIMESTAMP* field. The single space after *HOSTNAME* is part of the *HOSTNAME* field.
- *vendor* is the vendor name for the vendor-specific syslog messages or DOCSIS for the standard DOCSIS messages.
- *eventId* is an ASCII representation of the INTEGER number in decimal format, enclosed in angle brackets, which uniquely identifies the type of event. The CM MUST equate the eventId with the value stored in the docsDevEvId object in docsDevEventTable. For the standard DOCSIS events this number is converted from the error code using the following rules:
- The number is an eight-digit decimal number.
- The first two digits (left-most) are the ASCII code for the letter in the Error code.
- The next four digits are filled by 2 or 3 digits between the letter and the dot in the Error code with zero filling in the gap in the left side.

- The last two digits are filled by the number after the dot in the Error code with zero filling in the gap in the left side.

For example, event D04.2 is converted into 68000402, and Event I114.1 is converted into 73011401. This convention only uses a small portion of available number space reserved for DOCSIS (0 to 2^{31}-1). The first letter of an error code is always in upper-case. See Annex C for event definitions.

- *text* contains the textual description for the standard DOCSIS event message, as defined in Annex C.
- *vendor-specific-text* contains vendor-specific information. This field is optional.

For example, the syslog event for the event D04.2, "ToD Response received - Invalid data format", is as follows:

```
<132>CABLEMODEM[DOCSIS]: <68000402> ToD Response received - Invalid data format
```

The number 68000402 in the example is the number assigned by DOCSIS to this particular event.

The CM MAY report non-DOCSIS events in the standard syslog message format [RFC 3164] rather than the DOCSIS syslog message format defined above. When the CM sends a syslog message for an event not defined in this specification, the CM MAY send it according to the format and semantics of the elements defined above.

8.1.2.2 BIT Values for docsDevEvReporting (RFC 4639)

Permissible BIT values for [RFC 4639] docsDevEvReporting objects include:

1: local(0)

2: traps(1)

3: syslog(2)

4: localVolatile(8)

5: stdInterface(9)

Bit-0 means non-volatile Local Log storage and bit-8 is used for volatile Local Log storage (see Section 8.1.2.1). Bit-1 means SNMP Notifications which correspond to both SNMP Trap and SNMP Inform.

For backward compatibility with Pre-3.0 DOCSIS devices, the CM MUST support bit-3 in docsDevEvReporting BITS encoding for volatile Local Log storage.

DOCSIS 3.0 devices need to support bit override mechanisms during SNMP SET operations with either one-byte or two-byte BITS encoding for docsDevEvReporting for backward compatibility with Pre-3.0 DOCSIS behavior.

The CM MUST use the bit-3 value to set both bit-3 and bit-8 for SNMP SET operations on docsDevEvReporting using a one-byte BITS encoded value. Therefore, the CM reports bit-3 and bit-8 with identical values for SNMP GET operations.

The CM MUST use the bit-8 value to set both bit-3 and bit-8 for SNMP SET operations, irrespective of the bit-3 value, on docsDevEvReporting using a two or more byte BITS encoded value.

The CM MAY support bit-9 in docsDevEvReporting BITS encoding in accordance with [RFC 4639] definition.

A CM that reports an event by SNMP Notification or syslog MUST also report the event by a Local Log (volatile or non-volatile).

Combinations of docsDevEvReporting with traps(1) and/or syslog(2) bits with no Local Log bits (bit-0, bit-3 or bit-8) set are known as unacceptable combinations.

The CM MUST reject and report a 'Wrong Value' error for SNMPv2c/v3 PDUs or a 'Bad Value' error for SNMPv1 PDUs for any attempt to set docsDevEvReporting with unacceptable combinations.

The CM MUST accept any SNMP SET operation to docsDevEvReporting different than the unacceptable combinations.

The CM MUST ignore any undefined bits in docsDevEvReporting on SNMP SET operations and report a zero value for those bits.

Refer to Section 8.1.2.1.1 for details on Local Log requirements for the CM.

The CM MUST maintain the non-volatile storage when both non-volatile Local Log and volatile Local Log bits are set for a specific docsDevEvReporting event priority. The CM MAY maintain the volatile storage when both non-volatile Local Log and volatile Local Log bits are set for a specific docsDevEvReporting event priority. When both non-volatile Local Log and volatile Local Log bits are set for a specific docsDevEvReporting event priority, the CM MUST NOT report duplicate events in the docsDevEventTable.

8.1.2.3 Standard DOCSIS events for CMs

Aside from the procedures defined in this document, event recording conforms to the requirements of [RFC 4639]. Event descriptions are defined in English. A CM MUST implement event descriptors such that no event descriptor is longer than 255 characters, which is the maximum defined for SnmpAdminString [RFC 3411].

Events are identical if their EventIds are identical. For identical events occurring consecutively, the CM MAY choose to store only a single event. If a CM stores as a single event multiple identical events that occur consecutively, the CM MUST reflect in the event description the most recent event.

The EventId digit is a 32-bit unsigned integer. EventIds ranging [RFC 4639] from 0 to (2^{31} - 1) are reserved by DOCSIS. The CM MUST report in the docsDevEvTable [RFC 4639] the EventId as a 32-bit unsigned integer and convert the EventId from the error codes defined in Annex C to be consistent with this number format.

The DOCS-CABLE-DEVICE-MIB [RFC 4639] defines 8 priority levels and a corresponding reporting mechanism for each level.

Emergency event (priority 1)

Reserved for vendor-specific 'fatal' hardware or software errors that prevent normal system operation and cause the reporting system to reboot.

Every vendor may define their own set of emergency events. Examples of such events might be 'no memory buffers available', 'memory test failure', etc.

Alert event (priority 2)

A serious failure, which causes the reporting system to reboot, but it is not caused by hardware or software malfunctioning.

Critical event (priority 3)

A serious failure that requires attention and prevents the device from transmitting data, but could be recovered without rebooting the system. Examples of such events might be configuration file problems detected by the modem or the inability to get an IP address from the DHCP server.

Error event (priority 4)

A failure occurred that could interrupt the normal data flow, but will not cause the modem to re-register. Error events could be reported in real time by using the trap or syslog mechanism.

Warning event (priority 5)

A failure occurred that could interrupt the normal data flow, but will not cause the modem to re-register. 'Warning' level is assigned to events that both CM and CMTS have information about. To prevent sending the same event, both from the CM and the CMTS, the trap and syslog reporting mechanism is disabled by default for the CM for this level.

Notice event (priority 6)

The event is important, but is not a failure and could be reported in real time by using the trap or syslog mechanism. For a CM, an example of a Notice event is any event from 'SW UPGRADE SUCCESS' group.

Informational event (priority 7)

The event is of marginal importance, and is not failure, but could be helpful for tracing the normal modem operation.

Debug event (priority 8)

Reserved for vendor-specific non-critical events.

During CM initialization or reinitialization, the CM MUST support, as a minimum, the default event reporting mechanism shown in Table 15 - CM Default Event Reporting Mechanism Versus Priority.

The CM MAY implement default reporting mechanisms above the minimum requirements listed in Table 15 - CM Default Event Reporting Mechanism Versus Priority.

The reporting mechanism for each priority could be changed from the default reporting mechanism by using docsDevEvReporting object of DOCS-CABLE-DEVICE-MIB [RFC 4639].

The CM MUST populate the code of an event (as defined in Annex C) with Critical or Alert event priority through the docsIf3CmStatusCode SNMP object of DOCS-IF3-MIB before it recovers from the event. The CM MUST persist the docsIf3CmStatusCode across system reinitializations.

Table 15 - CM Default Event Reporting Mechanism Versus Priority

Event Priority	Local Log Non-volatile	SNMP Notification	Syslog	Local Log Volatile
Emergency	Yes	No	No	No
Alert	Yes	No	No	No
Critical	Yes	No	No	No
Error	No	Yes	Yes	Yes
Warning	No	No	No	Yes
Notice	No	Yes	Yes	Yes
Informational	No	No	No	No
Debug	No	No	No	No

The CM MUST format notifications that it generates for standard DOCSIS events as specified in Annex C.

8.1.2.4 Vendor-Specific Events

The CM MUST implement EventIds ranging from 2^{31} to (2^{32} - 1) as vendor-specific EventIds using the following format:

- Bit 31 is set to indicate vendor-specific event
- Bits 30-16 contain the lower 15 bits of the vendor's SNMP enterprise number
- Bits 15-0 are used by the vendor to number events

8.1.2.4.1 Event Priorities for DOCSIS and Vendor-specific Events

A CM MUST assign DOCSIS and vendor-specific events as indicated in Table 16 - Event Priority Assignment for CMs.

Table 16 - Event Priority Assignment for CMs

Event Priority	CM Event Assignment
Emergency	Vendor-specific
Alert	DOCSIS and Vendor-specific (optional*)
Critical	DOCSIS and Vendor-specific (optional*)
Error	DOCSIS and Vendor-specific (optional*)
Warning	DOCSIS and Vendor-specific (optional*)
Notice	DOCSIS and Vendor-specific (optional*)
Informational	DOCSIS and Vendor-specific (optional*)

Event Priority	CM Event Assignment
Debug	Vendor-specific
*Table Note: Vendor-specific optional event definitions are recommended only where the CM allows for sufficient storage of such events.	

8.1.3 Throttling, Limiting and Priority for Event, Trap and Syslog

8.1.3.1 Trap and Syslog Throttling, Trap and Syslog Limiting

A CM MUST support SNMP TRAP/INFORM and syslog throttling and limiting as described in DOCS-CABLE-DEVICE-MIB [RFC 4639], regardless of SNMP mode.

8.1.4 SNMPv3 Notification Receiver config file TLV

This section specifies processing requirements for the SNMPv3 Notification Receiver TLV [MULPIv4.0] when present in the configuration file. The SNMPv3 Notification Receiver TLV is used to configure SNMPv3 tables for notification transmission. The CM MUST process the SNMPv3 Notification Receiver TLV only if the CM is in SNMPv3 Coexistence Mode.

Based on the SNMPv3 Notification Receiver TLV, the CM MUST create entries in the following tables in order to cause the desired trap transmission:

- snmpNotifyTable
- snmpTargetAddrTable
- snmpTargetAddrExtTable
- snmpTargetParamsTable
- snmpNotifyFilterProfileTable
- snmpNotifyFilterTable
- snmpCommunityTable
- usmUserTable
- vacmContextTable
- vacmSecurityToGroupTable
- vacmAccessTable
- vacmViewTreeFamilyTable

The CM MUST NOT set to 'active' an entry created using the SNMPv3 Notification Receiver TLV (see the Common Radio Frequency Interface Encodings Annex of [MULPIv4.0]) which does not satisfy the corresponding [RFC 3413] requirements to do so. This type of misconfiguration doesn't stop the CM from registering; however, the SNMP notification process may not work as expected.

The mapping from the TLV to these tables is described in the following section.

8.1.4.1 Mapping of TLV Fields into Created SNMPv3 Table Rows

The following sections illustrate how the fields from the config file SNMPv3 Notification Receiver TLV elements are placed into the SNMPv3 tables. The TLV fields are shown below as:

Table 17 - SNMPv3 Notification Receiver TLV Mapping

Sub-TLVs	Variable Name	Associated MIB Object
SNMPv3 Notification Receiver IPv4 Address	TAddress	snmpTargetAddrTAddress [RFC 3413]
SNMPv3 Notification Receiver IPv6 Address	TAddress	snmpTargetAddrTAddress [RFC 3413]

Sub-TLVs	Variable Name	Associated MIB Object
SNMPv3 Notification Receiver UDP Port Number	Port	snmpTargetAddrTAddress [RFC 3413]
SNMPv3 Notification Receiver Trap Type	TrapType	see following sections
SNMPv3 Notification Receiver Timeout	Timeout	snmpTargetAddrTimeout [RFC 3413]
SNMPv3 Notification Receiver Retries	Retries	snmpTargetAddrRetryCount [RFC 3413]
SNMPv3 Notification Receiver Filtering Parameters	FilterOID	see following sections
SNMPv3 Notification Receiver Security Name	SecurityName	see following sections

The variable names from Table 17 are defined as follows:

<*TAddress*>A 32-bit IPv4 or IPv6 address of a notification receiver

<*Port*>A 16-bit UDP Port number on the notification receiver to receive the notifications

<*TrapType*>Defines the notification type as explained above

<*Timeout*>16-bit timeout, in milliseconds to wait before sending a retry of an Inform Notification

<*Retries*>16-bit number of times to retry an Inform after the first Inform transmission

<*FilterOID*>The OID of the snmpTrapOID value that is the root of the MIB subtree that defines all of the notifications to be sent to the Notification Receiver.

<*SecurityName*>The security name specified on the TLV element, or "@config" if not specified.

Table 18 through Table 29 are shown in the order that the agent will search down through them when a notification is generated in order to determine to whom to send the notification, and how to fill out the contents of the notification packet.

In configuring entries in these SNMPv3 tables, note the following:

The Community Name for traps in SNMPv1 and SNMPv2 packets is configured as "public". The Security Name in traps and informs in SNMPv3 packets where no security name has been specified is configured as "@config", in which case the security level is "noAuthNoPriv".

Several columnar objects are configured with a value beginning with the string "@config". If these tables are configured through other mechanisms, network operators should not use values beginning with "@config" to avoid conflicts with the mapping process specified here.

8.1.4.1.1 snmpNotifyTable

The snmpNotifyTable is defined in the "Notification MIB Module" section of [RFC 3413].

The CM MUST create two rows with fixed values if one or more SNMPv3 Notification Receiver TLV elements are present in the config file.

Table 18 - snmpNotifyTable

Column Name (* = Part of Index)	1st Row Column Value	2nd Row Column Value
* snmpNotifyName	"@config_inform"	"@config_trap"
snmpNotifyTag	"@config_inform"	"@config_trap"
snmpNotifyType	inform (2)	trap (1)
snmpNotifyStorageType	volatile (2)	volatile (2)
snmpNotifyRowStatus	active (1)	active (1)

8.1.4.1.2 snmpTargetAddrTable

The snmpTargetAddrTable is defined in the "Definitions" section of [RFC 3413].

The CM MUST create one row in snmpTargetAddrTable for each entry defined in Table 19 - snmpTargetAddrTable.

Thus, two entries are created in this table if both SNMPv3 Notification Receiver IPv4 Address and SNMPv3 Notification Receiver IPv6 Address sub-TLVs are included in the same TLV. All other parameters are the same.

Table 19 - snmpTargetAddrTable

Column Name (* = Part of Index)	Column Value
* snmpTargetAddrName	"@config_n_IPv[4 | 6]" where n is 0..m-1 and m is the number of SNMPv3 Notification Receiver config file TLVs @config_n_IPv4 is for an entry created if SNMPv3 Notification Receiver config file TLV contains <TrapType> of TDomain SnmpUDPAddress @config_n_IPv6 is for an entry created if SNMPv3 Notification Receiver config file TLV contains <TrapType> of TDomain TransportAddressIPv6
snmpTargetAddrTDomain	IPv4: snmpUDPDomain [RFC 3417] IPv6: transportDomainUdpIpv6 [RFC 3419]
snmpTargetAddrTAddress (IP Address and UDP Port of the Notification Receiver)	IPv4: SnmpUDPAddress [RFC 3417] OCTET STRING (6) Octets 1-4: <*TAddress*> Octets 5-6: <*Port*> IPv6: TransportAddressIPv6 [RFC 3419] OCTET STRING (18) Octets 1-16: <*TAddress*> Octets 17-18: <*Port*>
snmpTargetAddrTimeout	<Timeout>
snmpTargetAddrRetryCount	<*Retries*>
snmpTargetAddrTagList	"@config_trap" if <*TrapType*> is 1, 2, or 4 "@config_inform" if <*TrapType*> is 3 or 5
snmpTargetAddrParams	"@config_n"
snmpTargetAddrStorageType	volatile (2)
snmpTargetAddrRowStatus	active (1)

8.1.4.1.3 snmpTargetAddrExtTable

The snmpTargetAddrExtTable is defined in the "SNMP Community MIB Module" section of [RFC 3584].

The CM MUST create one row in snmpTargetAddrExtTable for each entry defined in Table 19 - snmpTargetAddrTable.

Table 20 - snmpTargetAddrExtTable

Column Name (* = Part of Index)	Column Value
* snmpTargetAddrName	"@config_n_IPv[4 | 6]" where n is 0..m-1 and m is the number of SNMPv3 Notification Receiver config file TLVs (see Table 19 for details).
snmpTargetAddrTMask	<Zero-length OCTET STRING>
snmpTargetAddrMMS	SM Maximum Message Size

8.1.4.1.4 snmpTargetParamsTable

The snmpTargetParamsTable is defined in the "Definitions" section of [RFC 3413].

The CM MUST create one row in snmpTargetParamsTable for each SNMPv3 Notification Receiver TLV in the config file.

Table 21 - snmpTargetParamsTable

Column Name (* = Part of Index)	Column Value
* snmpTargetParamsName	"@config_n" where n is 0..m-1 and m is the number of SNMPv3 Notification Receiver config file TLVs
snmpTargetParamsMPModel SYNTAX: SnmpMessageProcessingModel	SNMPv1 (0) if <*TrapType*> is 1 SNMPv2c (1) if <*TrapType*> is 2 or 3 SNMPv3 (3) if <*TrapType*> is 4 or 5

Column Name (* = Part of Index)	Column Value
snmpTargetParamsSecurityModel SYNTAX: SnmpSecurityModel	SNMPv1 (1) if <*TrapType*> is 1 SNMPv2c (2) if <*TrapType*> is 2 or 3 USM (3) if <*TrapType*> is 4 or 5 **Note:** The mapping of SNMP protocol types to value here are different from snmpTargetParamsMPModel
snmpTargetParamsSecurityName	If <*TrapType*> is 1, 2, or 3, or if the <*Security Name*> field is zero-length: "@config" If <*TrapType*> is 4 or 5, and the <*Security Name*> field is non-zero length: <*SecurityName*>
snmpTargetParamsSecurityLevel	If <*TrapType*> is 1, 2, or 3, or if the <*Security Name*> field is zero-length: noAuthNoPriv (1) If <*TrapType*> is 4 or 5, and the <*Security Name*> field is non-zero length: The security level of <*SecurityName*>
snmpTargetParamsStorageType	volatile (2)
snmpTargetParamsRowStatus	active (1)

8.1.4.1.5 snmpNotifyFilterProfileTable

The snmpNotifyFilterProfileTable is defined in the "Notification MIB Module" section of [RFC 3413].

The CM MUST create one row in snmpNotifyFilterProfileTable for each SNMPv3 Notification Receiver TLV that has a non-zero <*FilterOID*>.

Table 22 - snmpNotifyFilterProfileTable

Column Name (* = Part of Index)	Column Value
* snmpTargetParamsName	"@config_n" where n is 0..m-1 and m is the number of SNMPv3 Notification Receiver config file TLVs
snmpNotifyFilterProfileName	"@config_n" where n is 0..m-1 and m is the number of SNMPv3 Notification Receiver config file TLVs
snmpNotifyFilterProfileStorType	volatile (2)
snmpNotifyFilterProfileRowStatus	active (1)

8.1.4.1.6 snmpNotifyFilterTable

The snmpNotifyFilterTable is defined in the "Notification MIB Module" section of [RFC 3413].

The CM MUST create one row in snmpNotifyFilterTable for each SNMPv3 Notification Receiver TLV that has a non-zero <*FilterOID*>.

Table 23 - snmpNotifyFilterTable

Column Name (* = Part of Index)	Column Value
* snmpNotifyFilterProfileName	"@config_n" where n is 0..m-1 and m is the number of SNMPv3 Notification Receiver config file TLVs
* snmpNotifyFilterSubtree	<*FilterOID*>
snmpNotifyFilterMask	<Zero-length OCTET STRING>
snmpNotifyFilterType	included (1)
snmpNotifyFilterStorageType	volatile (2)
snmpNotifyFilterRowStatus	active (1)

8.1.4.1.7 snmpCommunityTable

The snmpCommunityTable is defined in the "SNMP Community MIB Module" section of [RFC 3584].

The CM MUST create one row in snmpCommunityTable with fixed values if one or more SNMPv3 Notification Receiver TLVs are present in the config file. This causes SNMPv1 and v2c notifications to contain the community string in snmpCommunityName.

Table 24 - snmpCommunityTable

Column Name (* = Part of Index)	Column Value
* snmpCommunityIndex	"@config"
snmpCommunityName	"public"
snmpCommunitySecurityName	"@config"
snmpCommunityContextEngineID	<the engineID of the cable modem>
snmpCommunityContextName	<Zero-length OCTET STRING>
snmpCommunityTransportTag	<Zero-length OCTET STRING>
snmpCommunityStorageType	volatile (2)
snmpCommunityStatus	active (1)

8.1.4.1.8 usmUserTable

The usmUserTable is defined in the "Definitions" section of [RFC 3414].

The CM MUST create one row in usmUserTable with fixed values if one or more SNMPv3 Notification Receiver TLVs are present in the config file. Other rows are created, one each time the engine ID of a trap receiver is discovered. This specifies the username on the remote notification receivers to which notifications are to be sent.

One row in the usmUserTable is created. When the engine ID of each notification receiver is discovered, the agent copies this row into a new row and replaces the 0x00 in the usmUserEngineID column with the newly-discovered value.

Table 25 - usmUserTable

Column Name (* = Part of Index)	Column Value
* usmUserEngineID	0x00
* usmUserName	"@config" When other rows are created, this is replaced with the <*SecurityName*> field from the SNMPv3 Notification Receiver config file TLV.
usmUserSecurityName	"@config" When other rows are created, this is replaced with the <*SecurityName*> field from the SNMPv3 Notification Receiver config file TLV.
usmUserCloneFrom	<don't care> This row cannot be cloned.
usmUserAuthProtocol	None When other rows are created, this is replaced with None or MD5, depending on the security level of the V3 User.
usmUserAuthKeyChange	<don't care> Write-only
usmUserOwnAuthKeyChange	<don't care> Write-only
usmUserPrivProtocol	None When other rows are created, this is replaced with None or DES, depending on the security level of the V3 User.
usmUserPrivKeyChange	<don't care> Write-only
usmUserOwnPrivKeyChange	<don't care> Write-only
usmUserPublic	<Zero-length OCTET STRING>
usmUserStorageType	volatile (2)
usmUserStatus	active (1)

8.1.4.1.9 vacmContextTable

The vacmContextTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create one row in vacmContextTable with the zero length octet string for vacmContextName object.

Table 26 - vacmContextTable

Column Name (* = Part of Index)	Column Value
* vacmContextName	<Zero-length OCTET STRING>

8.1.4.1.10 vacmSecurityToGroupTable

The vacmSecurityToGroupTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create three rows in vacmSecurityToGroupTable with fixed values if one or more SNMPv3 Notification Receiver TLVs are present in the config file.

Table 27 depicts the three rows with fixed values which are used for the SNMPv3 Notification Receiver TLV entries with <*TrapType*> set to 1, 2, or 3, or with a zero-length <*SecurityName*>. The SNMPv3 Notification Receiver TLV entries with <*TrapType*> set to 4 or 5 and a non-zero length <*SecurityName*> will use the rows created in the vacmSecurityToGroupTable by the DH Kickstart process.

Table 27 - vacmSecurityToGroupTable

Column Name (* = Part of Index)	First Row Column Value	Second Row Column Value	Third Row Column Value
* vacmSecurityModel	SNMPV1 (1)	SNMPV2c (2)	USM (3)
* vacmSecurityName	"@config"	"@config"	"@config"
vacmGroupName	"@configV1"	"@configV2"	"@configUSM"
vacmSecurityToGroupStorageType	volatile (2)	volatile (2)	volatile (2)
vacmSecurityToGroupStatus	active (1)	active (1)	active (1)

8.1.4.1.11 vacmAccessTable

The vacmAccessTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create three rows in vacmAccessTable with fixed values if one or more SNMPv3 Notification Receiver TLVs are present in the config file.

Table 28 depicts the three rows with fixed values which are used for the SNMPv3 Notification Receiver TLV entries with <*TrapType*> set to 1, 2, or 3, or with a zero-length <*SecurityName*>. The SNMPv3 Notification Receiver TLV entries with <*TrapType*> set to 4 or 5 and a non-zero length <*SecurityName*> will use the rows created in the vacmAccessTable by the DH Kickstart process.

Table 28 - vacmAccessTable

Column Name (* = Part of Index)	First Row Column Value	Second Row Column Value	Third Row Column Value
* vacmGroupName	"@configV1"	"@configV2"	"@configUSM"
* vacmAccessContextPrefix	<zero-length string>	<zero-length string>	<zero-length string>
* vacmAccessSecurityModel	SNMPV1 (1)	SNMPV2c (2)	USM (3)
* vacmAccessSecurityLevel	noAuthNoPriv (1)	noAuthNoPriv (1)	noAuthNoPriv (1)
vacmAccessContextMatch	exact (1)	exact (1)	exact (1)
vacmAccessReadViewName	<Zero-length OCTET STRING>	<Zero-length OCTET STRING>	<Zero-length OCTET STRING>
vacmAccessWriteViewName	<Zero-length OCTET STRING>	<Zero-length OCTET STRING>	<Zero-length OCTET STRING>
vacmAccessNotifyViewName	"@config"	"@config"	"@config"

Column Name (* = Part of Index)	First Row Column Value	Second Row Column Value	Third Row Column Value
vacmAccessStorageType	volatile (2)	volatile (2)	volatile (2)
vacmAccessStatus	active (1)	active (1)	active (1)

8.1.4.1.12 vacmViewTreeFamilyTable

The vacmViewTreeFamilyTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create one row in vacmViewTreeFamilyTable with fixed values if one or more SNMPv3 Notification Receiver TLVs are present in the config file.

This row is used for the SNMPv3 Notification Receiver TLV entries with <*TrapType*> set to 1, 2, or 3 or with a zero-length <*SecurityName*>. The SNMPv3 Notification Receiver TLV entries with <*TrapType*> set to 4 or 5 and a non-zero length <*SecurityName*> will use the rows created in the vacmViewTreeFamilyTable by the DH Kickstart process.

Table 29 - vacmViewTreeFamilyTable

Column Name (* = Part of Index)	Column Value
* vacmViewTreeFamilyViewName	"@config"
* vacmViewTreeFamilySubtree	1.3
vacmViewTreeFamilyMask	<default from MIB>
vacmViewTreeFamilyType	included (1)
vacmViewTreeFamilyStorageType	volatile (2)
vacmViewTreeFamilyStatus	active (1)

8.1.5 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), and trace route (UDP and various ICMP Destination Unreachable flavors). The CM MUST respond to ICMP Echo Request (ping) messages received through its CMCI [CMCIv3.0] interface(s) to enable local connectivity testing from a subscriber's PC to the modem. The CM MUST support IP end-station generation of ICMP error messages and processing of all ICMP messages.

Syslog requirements are defined in Section 8.1.2.

8.2 Configuration Management

The CM is required to support the SNMP protocol interface as specified in Section 6. Section 7 defines the SNMP MIB objects that are required to be supported by a CM.

In addition to the SNMP interface to modify the attribute values stored in the CM, vendors may specify additional methods such as Command Line Interface (CLI) or an HTTP interface, as examples. Irrespective of the method used, it is necessary to assure the data integrity as a result of changes performed using different interfaces. For example, when the attribute value is modified using one management interface, this changed value is reported when that attribute is accessed from any of the other interfaces. When a change in the value of the attribute does not succeed, requesting the same change from another interface should also results in failure (assuming the same level of access control for all those interfaces for the specific operation). If an event is generated as a result of making the change in one management interface, this is reported independent of which method was used to initiate the change.

8.2.1 Version Control

The CM MUST support software revision and operational parameter configuration interrogation.

The CM includes the hardware version, boot ROM image version, vendor name, current software version, and model number in the sysDescr object (from [RFC 3418]).

The CM MUST support docsDevSwCurrentVers MIB object (from [RFC 4639]) and report the current software version of the CM.

The CM MUST report for the sysDescr object the Type and Value fields identified in Table 30 - sysDescr Format.

Table 30 - sysDescr Format

Type	Value
HW_REV	<Hardware Version>
VENDOR	<Vendor Name>
BOOTR	<Boot ROM Version>
SW_REV	<Software Version>
MODEL	<Model Number>

The CM MUST report each Type and Value for the sysDescr object identified in Table 30 - sysDescr Format, with each Type field and corresponding Value field separated with a colon followed by a single blank space and each Type-Value pair is separated by a semicolon followed by a single blank space. The correct format is illustrated below.

```
HW_REV: <value>; VENDOR: <value>; BOOTR: <value>; SW_REV: <value>; MODEL: <value>
```

For instance, a sysDescr of a CM of vendor X, hardware version 5.2, boot ROM image version 1.4, software version 2.2, and model number Z is formatted as follows:

any text<<HW_REV: 5.2; VENDOR: X; BOOTR: 1.4; SW_REV: 2.2; MODEL: Z>>any text

The CM MUST report all of the information necessary in determining what software the CM is capable of being upgraded to. If any fields in Table 30 - sysDescr Format are not applicable, the CM MUST report "NONE" as the value.

For instance, a sysDescr of a CM of vendor X, hardware version 5.2, no boot ROM image information, software version 2.2, and model number Z is formatted as follows:

any text<<HW_REV: 5.2; VENDOR: X; BOOTR: NONE; SW_REV: 2.2; MODEL: Z>>any text

The intent of specifying the format of sysDescr is to define how to report information in a consistent manner so that sysDescr field information can be programmatically parsed. This format specification does not intend to restrict the vendor's hardware version numbering policy.

8.2.2 System Configuration

The CM MUST support system configuration by configuration file, configuration-file-based SNMP encoded object, and SNMP Set operation. The CM MUST support any valid configuration file created in accordance with configuration file size limitations defined in the CM Configuration Interface Specification Annex in [MULPIv4.0].

8.2.3 Secure Software Download

The CM Secure Software Download (SSD) process is documented in detail in the Secure Software Download section of [SECv4.0].

The CM MUST use the Secure Software Download mechanism to perform software upgrade regardless of the DOCSIS specification version the CMTS it is connected to complies with.

There are two available Secure Software Download schemes: the manufacturer control scheme and the operator control scheme.

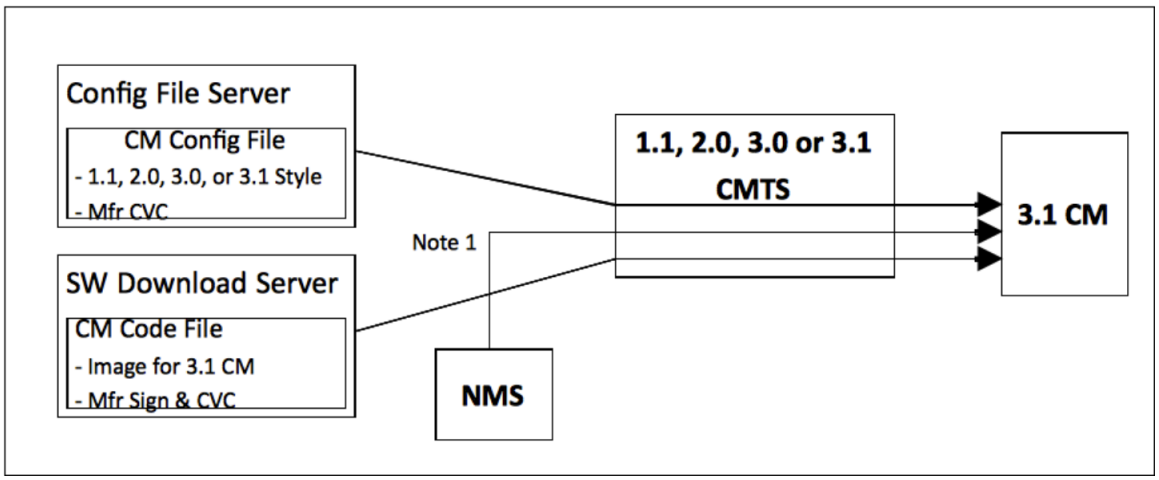


Figure 6 - Manufacturer Control Scheme

In reference to Figure 6 above:

Note 1: Use docsDevSoftware group ([RFC 2669], [RFC 4639]) in case that the software downloading is triggered by the MIB.

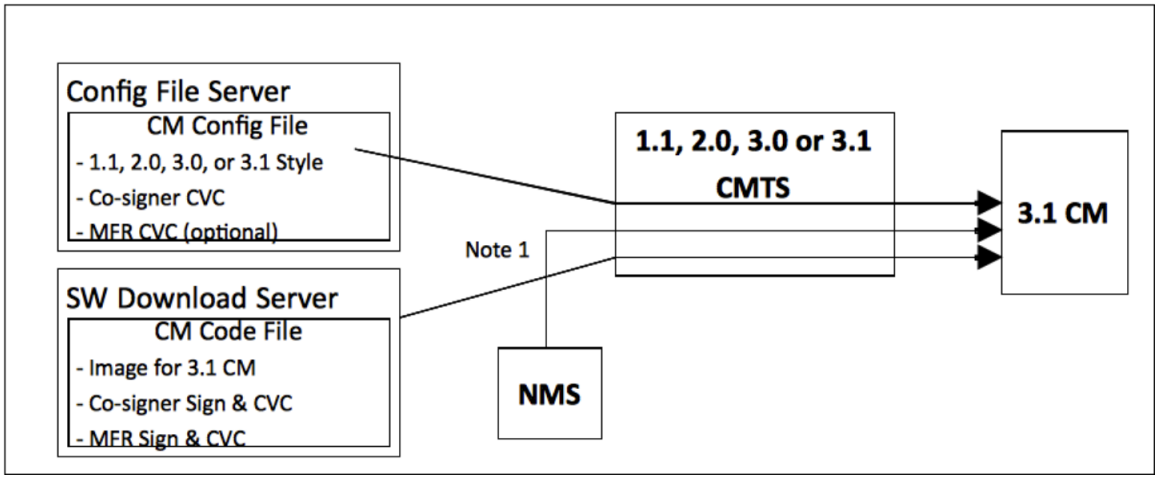


Figure 7 - Operator Control Scheme

In reference to Figure 7 above:

Note 1: Use docsDevSoftware group ([RFC 2669], [RFC 4639]) in case that the software downloading is triggered by the MIB.

Prior to Secure Software Download initialization, CVC information needs to be initialized at the CM for software upgrade. Depending on the scheme (described above) that the operator chooses to implement, the CM requires appropriate CVC information in the configuration file. It is recommended that CVC information always be present in the configuration file so that a device will always have the CVC information initialized and read if the operator decides to use a SNMP-initiated upgrade as a method to trigger a Secure Software Download operation. If the operator decides to use a configuration-file-initiated upgrade as a method to trigger Secure Software Download, CVC information needs to be present in the configuration file at the time the CM is rebooted to get the configuration file that will trigger the upgrade only.

There are two methods to trigger Secure Software Download: SNMP-initiated and configuration-file-initiated. The CM MUST support both SNMP-initiated and configuration-file-initiated methods to trigger Secure Software Download.

The following describes the SNMP-initiated mechanism. Prior to an SNMP-initiated upgrade, a CM MUST have valid X.509-compliant code verification certificate information. From a network management station:

1. Set docsDevSwServerAddressType to 'ipv4' or 'ipv6'.
2. Set docsDevSwServerAddress to the IPv4 or IPv6 address of the Software Download server for software upgrades.
3. Set docsDevSwFilename to the file path name of the software upgrade image.
4. Set docsDevSwAdminStatus to 'upgradeFromMgt'.

If docsDevSwAdminStatus is set to 'ignoreProvisioningUpgrade', the CM MUST ignore any software download configuration file setting and not attempt a configuration file initiated upgrade.

The CM MUST preserve the value of docsDevSwAdminStatus across reset/reboots until over-written from an SNMP manager or by a TLV-11 [MULPIv4.0] setting in the CM configuration file. That is, the value of docsDevSwAdminStatus is required to persist across CM reboots.

The CM MUST report 'allowProvisioningUpgrade' as the default value of docsDevSwAdminStatus until it is over-written by 'ignoreProvisioningUpgrade', following a successful SNMP-initiated software upgrade or otherwise altered by the management station.

The CM MUST preserve the value of docsDevSwOperStatus across reset/reboots. That is, the value of the CM's docsDevSwOperStatus object is required to persist across resets to report the outcome of the last software upgrade attempt.

After the CM has completed a configuration-file-initiated secure software upgrade, the CM MUST reboot and become operational with the correct software image as specified in [MULPIv4.0]. After the CM is registered following a reboot after a configuration file initiated secure software upgrade, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MAY report the filename of the software currently operating on the CM as the value for docsDevSwFilename.
- The CM MAY report the IP address of the Software Download server containing the software that is currently operating on the CM as the value for docsDevSwServerAddress.
- The CM MUST report 'completeFromProvisioning' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of the software that is operating on the CM as the value for docsDevSwCurrentVers.

After the CM has completed an SNMP-initiated secure software upgrade, the CM MUST reboot and become operational with the correct software image as specified in [MULPIv4.0]. After the CM is registered following a reboot after an SNMP-initiated secure software upgrade, the CM MUST adhere to the following requirements:

- The CM MUST report 'ignoreProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MAY report the IP address of the Software Download server containing the software that is currently operating on the CM as the value for docsDevSwServerAddress.
- The CM MUST report 'completeFromMgt' as the value for docsDevOperStatus.
- The CM MUST report the current version of the software that is operating on the CM as the value for docsDevSwCurrentVers.

If the value of docsDevSwAdminStatus is 'ignoreProvisioningUpgrade', the CM MUST ignore any software upgrade value that is optionally included in the CM configuration file and become operational with the current software image after the CM is registered. After the CM is registered following a reboot with a software upgrade value in the CM configuration file, the CM MUST adhere to the following requirements:

- The CM MUST report 'ignoreProvisioningUpgrade' as the value for docsDevSwAdminStatus.

- The CM MAY report the filename of the software currently operating on the CM as the value for docsDevSwFilename.
- The CM MAY report the IP address of the Software Download server containing the software that is currently operating on the CM as the value for docsDevSwServerAddress.
- The CM MUST report 'completeFromMgt' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of the software that is operating on the CM as the value for docsDevSwCurrentVers.

Retries due to a power loss or reset are only required for an SNMP-initiated upgrade. If a power loss or reset occurs during a configuration-file-initiated upgrade, the CM will follow the upgrade TLV directives in the configuration file upon reboot. It will not retry the previous upgrade. The config file upgrade TLVs essentially provides a retry mechanism that is not available for an SNMP-initiated upgrade.

If a CM suffers a loss of power or resets during an SNMP-initiated upgrade, the CM MUST resume the upgrade without requiring manual intervention. When the CM resumes the upgrade process after a reset that occurred during an SNMP-initiated software upgrade, the CM MUST adhere to the following requirements:

- The CM MUST report 'upgradeFromMgt' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename of the software image to be upgraded as the value for docsDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software upgrade image to be upgraded as the value for docsDevSwServerAddress.
- The CM MUST report 'inProgress' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

In the case where the CM reaches the maximum number of TFTP Download Retries, as specified in the Parameters and Constraints Annex of [MULPIv4.0], resulting from multiple losses of power or resets during an SNMP-initiated upgrade, the CM MUST behave as specified in [MULPIv4.0]. In this case, after the CM is registered, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename of the software that failed the upgrade process as the value for docsDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software that failed the upgrade process as the value for docsDevSwServerAddress.
- The CM MUST report 'other' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

When the CM reboots following a reset that occurred during a configuration file-initiated software download, the CM MUST ignore the fact that a previous upgrade was in progress and either not perform an upgrade if no upgrade TLVs are present in the config file, or if upgrade TLVs are present, take the action described in the requirements in the section "Downloading Cable Modem Operating Software" of [MULPIv4.0], at the time of the reboot.

In the case where the CM had a configuration-file-initiated upgrade in progress during a reset and if there are no upgrade TLVs in the config file upon reboot, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MAY report the filename of the current software image as the value for docsDevSwFilename.
- The CM MAY report the IP address of the Software Download server containing the software that is currently operating in the CM as the value for docsDevSwServerAddress.

- The CM MUST report 'other' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

In the case where the CM had a configuration-file-initiated upgrade in progress during a reset, if there are upgrade TLVs in the config file upon reboot, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename contained in TLV-9 of the config file as the value for docsDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software to be loaded into the CM as the value for docsDevSwServerAddress, per the requirements stated in the section "Downloading Cable Modem Operating Software" of [MULPIv4.0].
- The CM MUST report 'inProgress' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

If a CM exhausts the required number of TFTP Request Retries, as specified in the Parameters and Constraints Annex of [MULPIv4.0], the CM MUST behave as specified in [MULPIv4.0]. If a CM exhausts the maximum number of configured TFTP Request Retries without successfully downloading the specified file, the CM MUST fall back to last known working image and proceed to an operational state. After a CM falls back to the last known working software image after exhausting the maximum number of configured TFTP Request Retries without successfully downloading the specified file, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename of the software that failed the upgrade process as the value for docDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software that failed the upgrade process as the value for docsDevSwServerAddress.
- The CM MUST report 'failed' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

In the case where a CM successfully downloads (or detects during download) an image that is not intended for the CM device, the CM behaves as specified in the section "Downloading Cable Modem Operating Software" of [MULPIv4.0]. If a CM successfully downloads an image that is not intended for it, or detects during the download of a software image that the image is not for itself, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename of the software that failed the upgrade as the value for docsDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software that failed the upgrade process as the value for docsDevSwServerAddress.
- The CM MUST report 'other' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

In the case where the CM determines that the download image is damaged or corrupted, the CM MUST reject the newly downloaded image. The CM MAY re-attempt to download if the maximum number of TFTP Download Retries has not been reached, as specified in the Parameters and Constants Annex of [MULPIv4.0]. If the CM chooses not to retry, the CM MUST fall back to the last known working image and proceed to an operational state

and generate appropriate event notification as specified in Annex C. If the CM does not retry to download a corrupted software image and falls back to the last known working software image, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename of the software that failed the upgrade as the value for docsDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software that failed the upgrade process as the value for docsDevSwServerAddress.
- The CM MUST report 'other' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

In the case where the CM determines that the image is damaged or corrupted, the CM MAY re-attempt to download the new image if the maximum number of TFTP Download Retries has not been reached, as specified in Parameters and Constraints Annex of [MULPIv4.0]. On the final consecutive failed retry of the CM software download attempt, the CM MUST fall back to the last known working image and proceed to an operational state and generate appropriate event notification as specified in Annex C. If a CM falls back to the last known working software image after failing the defined consecutive retry attempts, the CM MUST send two notifications, one to notify that the max retry limit has been reached, and another to notify that the image is damaged. Immediately after the CM reaches the operational state after failing the defined consecutive retry attempts to download a software image and falling back to the last known working software image, the CM MUST adhere to the following requirements:

- The CM MUST report 'allowProvisioningUpgrade' as the value for docsDevSwAdminStatus.
- The CM MUST report the filename of the software that failed the upgrade as the value for docsDevSwFilename.
- The CM MUST report the IP address of the Software Download server containing the software that failed the upgrade process as the value for docsDevSwServerAddress.
- The CM MUST report 'other' as the value for docsDevSwOperStatus.
- The CM MUST report the current version of software that is operating on the CM as the value for docsDevSwCurrentVers.

8.2.4 CM configuration files, TLV-11 and MIB OIDs/values

The following sections define the use of CM configuration file TLV-11 elements and the CM rules for translating TLV-11 elements into SNMP PDU (SNMP MIB OID/instance and MIB OID/instance value combinations; also referred to as SNMP varbinds).

This section also defines the CM behaviors, or state transitions, after either pass or fail of the CM configuration process.

For TLV-11 definitions, refer to the Common Radio Frequency Interface Encodings Annex of [MULPIv4.0].

8.2.4.1 CM configuration file TLV-11 element translation (to SNMP PDU)

TLV-11 translation defines the process used by the CM to convert CM configuration file information (TLV-11 elements) into SNMP PDU (varbinds). The CM is required to translate CM configuration file TLV-11 elements into a single SNMP PDU containing (n) MIB OID/instance and value components (SNMP varbinds). Once a single SNMP PDU is constructed, the CM processes the SNMP PDU and determines the CM configuration pass/fail based on the rules for CM configuration file processing, described below. However, if a CM is not physically capable of processing a potentially large single CM configuration file-generated SNMP PDU, the CM is still required to behave as if all MIB OID/instance and value components (SNMP varbinds) from CM configuration file TLV-11 elements are processed as a single SNMP PDU.

In accordance with [RFC 3416], the single CM configuration file generated SNMP PDU will be treated "as if simultaneous" and the CM MUST behave consistently, regardless of the order in which TLV-11 elements appear in the CM configuration file, or SNMP PDU.

The CM configuration file MUST NOT contain duplicate TLV-11 elements (duplicate means SNMP MIB object has identical OID). If the configuration file received by the CM contains duplicate TLV-11 elements, the CM MUST reject the configuration file.

8.2.4.1.1 Rules for CreateAndGo and CreateAndWait

The CM MUST support 'createAndGo' [RFC 2579] for row creation.

The CM MAY support 'createAndWait' [RFC 2579]. If the CM supports 'createAndWait', there is the constraint that CM configuration file TLV-11 elements MUST NOT be duplicated (all SNMP MIB OID/instance are required to be unique). If a CM constructs an SNMP PDU from a CM configuration file TLV-11 element that contains an SNMP 'createAndWait' value for a given SNMP MIB OID/instance, the CM MUST NOT also include in that SNMP PDU an SNMP Active value for the same SNMP MIB OID/instance (and vice versa). A CM MAY accept a configuration file that contains a TLV-11 'createAndWait' element if the intended result is to create an SNMP table row which will remain in the SNMP 'notReady' or SNMP 'notInService' state until a non-configuration file SNMP PDU is issued, from an SNMP manager, to update the SNMP table row status.

Both SNMP 'notReady' and SNMP 'notInService' states are valid table row states after an SNMP 'createAndWait' instruction.

8.2.4.2 CM configuration TLV-11 elements not supported by the CM

If any CM configuration file TLV-11 elements translate to SNMP MIB OIDs that are not MIB OID elements supported by the CM, then the CM MUST ignore those SNMP varbinds, and treat them as if they had not been present, for the purpose of CM configuration. This means that the CM will ignore SNMP MIB OIDs for other vendors' private MIBs as well as standard MIB elements that the CM does not support.

CMs that do not support SNMP CreateAndWait for a given SNMP MIB table MUST ignore, and treat as if not present, the set of columns associated with the SNMP table row.

If any CM configuration file TLV-11 element(s) are ignored, then the CM MUST report them via the CM configured notification mechanism(s), after the CM is registered. The CM MUST report ignored configuration file TLV-11 elements following the notification method in accordance with Section 8.1.2.3.

8.2.4.3 CM State after CM Configuration File Processing Success

After successful CM configuration via CM configuration file, the CM MUST proceed to register with the CMTS and proceed to its operational state.

8.2.4.4 CM State after CM Configuration File Processing Failure

If any CM configuration file generated SNMP PDU varbind performs an illegal set operation (illegal, bad, or inconsistent value) to any MIB OID/instance supported by the CM, the CM MUST reject the configuration file. The CM MUST NOT proceed with CM registration if it fails to download and process the configuration file.

8.3 Accounting Management

[CCAP-OSSIv4.0] defines an accounting management interface for subscriber usage-based applications denominated Subscriber Account Management Interface Specification (SAMIS). SAMIS is defined to enable prospective vendors of Cable Modems and Cable Modem Termination Systems to address the operational requirements of subscriber account management in a uniform and consistent manner.

8.3.1 Subscriber Usage Billing and Class of Services

The [MULPIv4.0] specification uses the concept of class of service as the term to indicate the type of data services a CM requests and receives from the CMTS; (see [MULPIv4.0]). From a high-level perspective, classes of services

are observed as subscriber types (e.g., residential or business) and the DOCSIS RFI MAC layer parameters fulfill the subscriber service needs.

8.3.1.1 DOCSIS 1.1 Quality of Service (QoS)

The [MULPIv4.0] specification provides a mechanism for a CM to register with its CMTS and to configure itself based on external QoS parameters when it is powered up or reset.

To quote (in part) from the Theory of Operation section of [MULPIv4.0]:

The principal mechanism for providing enhanced QoS is to classify packets traversing the RF MAC interface into a Service Flow. A Service Flow is a unidirectional flow of packets that provide a particular Quality of Service. The CM and the CMTS provide this QoS by shaping, policing, and prioritizing traffic according to the QoS Parameter Set defined for the Service Flow.

The requirements for Quality of Service include:

- A configuration and registration function for pre-configuring CM-based QoS Service Flows and traffic parameters.
- Utilization of QoS traffic parameters for downstream Service Flows.
- Classification of packets arriving from the upper layer service interface to a specific active Service Flow.
- Grouping of Service Flow properties into named Service Classes, so upper layer entities and external applications (at both the CM and the CMTS) can request Service Flows with desired QoS parameters in a globally consistent way.

A Service Class Name (SCN) is defined in the CMTS by provisioning (see Annex G. An SCN provides an association to a QoS Parameter Set. Service Flows that are created using an SCN are considered to be "named" Service Flows. The SCN identifies the service characteristics of a Service Flow to external systems such as a billing system or customer service system. For consistency in billing, operators should ensure that SCNs are unique within an area serviced by the same BSS that utilizes this interface. A descriptive SCN might be something like PrimaryUp, GoldUp, VoiceDn, or BronzeDn to indicate the nature and direction of the Service Flow to the external system.

A Service Package implements a Service Level Agreement (SLA) between the MSO and its Subscribers on the RFI interface. A Service Package might be known by a name such as Gold, Silver, or Bronze. A Service Package is itself implemented by the set of named Service Flows (using SCNs) that are placed into a CM Configuration File that is stored on a Config File server. **Note:** The CM Configuration File contains several kinds of information needed to properly configure the CM and its relationship with the CMTS, but for the sake of this discussion, only the Service Flow and Quality of Service components are of interest. The set of Service Flows defined in the CM Config File are used to create active Service Flows when the CM registers with the CMTS. Note that many Subscribers are assigned to the same Service Package and, therefore, many CMs use the same CM Config File to establish their active Service Flows.

A Service Package has to define at least two Service Flows known as Primary Service Flows that are used by default when a packet matches none of the classifiers for the other Service Flows. A CM Config File that implements a Service Package, therefore, needs to define the two primary Service Flows using SCNs (e.g., PrimaryUp and PrimaryDn) that are known to the CMTS if these Service Flows are to be visible to external systems by this billing interface. Note that it is often the practice in a usage sensitive billing environment to segregate the operator's own maintenance traffic, to and from the CM, into the primary service flows so that this traffic is not reflected in the traffic counters associated the subscriber's SLA service flows.

The [MULPIv4.0] specification also provides for dynamically created Service Flows. An example could be a set of dynamic Service Flows created by an embedded PacketCable Multimedia Terminal Adapter (MTA) to manage VoIP signaling and media flows. All dynamic Service Flows need to be created using an SCN known to the CMTS if they are to be visible to the billing system. These dynamic SCNs do not need to appear in the CM Config File but the MTA may refer to them directly during its own initialization and operation.

During initialization, a CM communicates with a DHCP Server that provides the CM with its assigned IP address and, in addition, receives a pointer to the Config File server that stores the assigned CM Config File for that CM. The CM reads the CM Config File and forwards the set of Service Flow definitions (using SCNs) up to the CMTS.

The CMTS then performs a macro-expansion on the SCNs (using its provisioned SCN templates) into QoS Parameter Sets sent in the Registration Response for the CM. Internally, each active Service Flow is identified by a 32-bit SFID assigned by the CMTS to a specific CM (relative to the RFI interface). For billing purposes, however, the SFID is not sufficient as the only identifier of a Service Flow because the billing system cannot distinguish the class of service being delivered by one SFID from another. Therefore, the SCN is necessary, in addition to the SFID, to identify the Service Flow's class of service characteristics to the billing system.

The billing system can then rate the charges differently for each of the Service Flow traffic counts based on its Service Class (e.g., Gold octet counts are likely to be charged more than Bronze octet counts). Thus, the billing system obtains, from the CMTS, the traffic counts for each named Service Flow (identified by SFID and SCN) that a subscriber's CM uses during the billing data collection interval. This is true even if multiple active Service Flows (i.e., SFIDs) are created using the same SCN for a given CM over time. This will result in multiple billing records for the CM for Service Flows that have the same SCN (but different SFIDs). Note that the SFID is the primary key to the Service Flow. When an active Service Flow exists across multiple sequential billing files, the SFID allows the sequence of recorded counter values to be correlated to the same Service Flow instance.

8.4 Performance Management

At the CATV MAC and PHY layers, 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 interface statistics [RFC 2863] and service queue statistics (from [RFC 4546] and Annex G).

At the LLC layer, the performance management focus is on bridge traffic management. The CM implements the Bridge MIB [RFC 4188] as specified in Section 7.1.3.10 and Annex A.

The DOCS-IF-MIB [RFC 4546] 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 DOCS-IF-MIB [RFC 4546] also includes counters to track MAC state, such as collisions and excessive retries for requests, immediate data transmits, and initial ranging requests. Annex D provides Proactive Network Maintenance monitoring and diagnostic capabilities for detecting cable plant issues.

A final performance concern is the ability to diagnose unidirectional loss. The CM implements the MIB-II [RFC 1213] Interfaces Group [RFC 2863] as specified in Section 7.1.3.8 and Annex A.

8.4.1 Treatment and Interpretation of MIB Counters

Octet and packet counters implemented as Counter32 and Counter64 MIB objects are monotonically increasing positive integers with no specific initial value and a maximum value based on the counter size that will roll-over to zero when it is exceeded. In particular, counters are defined such that the only meaningful value is the difference between counter values as seen over a sequence of counter polls. However, there are two situations that can cause this consistent monotonically increasing behavior to change: 1) resetting the counter due to a system or interface reinitialization or 2) a rollover of the counter when it reaches its maximum value of 2^{32}-1 or 2^{64}-1. In these situations, it needs to be clear what the expected behavior of the counters should be.

Case 1: The state of an interface changes resulting in an "interface counter discontinuity" as defined in [RFC 2863].

In the case where the state of an interface within the CM changes resulting in an "interface counter discontinuity" [RFC 2863], the CM value of the ifXTable.ifXEntry.ifCounterDiscontinuityTime for the affected interface MUST be set to the current value of sysUpTime and ALL counters for the affected interface set to ZERO. When setting the ifAdminStatus of the affected interface to down(2), the CM MUST NOT consider this as an interface reset.

Case 2: SNMP Agent Reset.

An SNMP Agent Reset is defined as the reinitialization of the SNMP Agent software caused by a device reboot or device reset initiated through SNMP.

In the case of an SNMP Agent Reset within the CM, the CM MUST:

- set the value of sysUpTime to zero (0)
- set all interface ifCounterDiscontinuityTime values to zero (0)

- set all interface counters to zero (0)
- set all other counters maintained by the CM SNMP Agent to zero (0).

Case 3: Counter Rollover.

When a Counter32 object within the CM reaches its maximum value of 4,294,967,295, the next value MUST be ZERO. When a Counter64 object within the CM reaches its maximum value of 18,446,744,073,709,551,615, the next value MUST be ZERO.

Note: Unless a CM vendor provides a means outside of SNMP to preset a Counter64 or Counter32 object to an arbitrary value, it will not be possible to test any rollover scenarios for Counter64 objects (and many Counter32 objects as well). This is because it is not possible for these counters to rollover during the service life of the device (see discussion in section 3.1.6 of [RFC 2863]).

8.5 Security Management

The CM is required to provide SNMP responses in accordance with the SNMP framework defined in [RFC 3411] through [RFC 3416] and the guidelines defined in this section.

8.5.1 CM SNMP Modes of Operation

The CM MUST support SNMPv1, SNMPv2c, and SNMPv3 as well as SNMP-coexistence [RFC 3584] subject to the requirements in the following sections.

The CM access control configuration supports SNMPv1v2c in NmAccess mode and SNMPv1v2c Coexistence mode as described in [RFC 4639] and Section 8.5.2.7, respectively.

8.5.2 CM SNMP Access Control Configuration

The CM SNMP access control is configured via the CM config file and later updated for an authorized entity. The confidentiality and authenticity of the information in the config file is defined in [MULPIv4.0] and [SECv4.0]. The CM access control configuration supports SNMPv3 configuration through the Diffie-Hellman SNMP Kickstart process defined in Section 8.5.2.6.

8.5.2.1 SNMP Operation Before CM Registration

IP connectivity between the CM and the SNMP management station MUST be implemented as described in Section 9.1.

The CM MUST provide read-only access to the following MIB objects prior to CM registration:

- docsIfDownChannelFrequency
- docsIfDownChannelPower
- docsIf3CmStatusValue
- docsDevEventTable

The CM MAY provide read-only access to the following MIB objects prior to CM registration:

- sysDescr
- sysUptime
- ifXTable
- docsIfUpChannelFrequency
- docsIfSignalQualityTable
- docsIfCmCmtsAddress
- docsIfCmStatusUsTxPower

- docsDevSwCurrentVers

The CM MUST NOT provide access to the following information prior to CM registration:

- QoS service flow information
- Configuration file contents
- Secure Software Download information
- Key authentication and encryption material
- SNMP management and control
- DOCSIS functional modules statistics and configuration
- Network provisioning hosts and servers IPs addresses

Additionally, prior to registration, the CM MUST adhere to the following requirements:

- The CM MAY provide access to additional information not listed in the statements above.
- The CM MUST NOT provide SNMP access from the RF interface prior to registration.
- The CM MUST accept any SNMPv1/v2c packets regardless of SNMP community string.
- The CM MUST drop all SNMPv3 packets.

The CM MUST NOT complete registration prior to successful processing of all MIB elements in the configuration file.

The CM MUST complete registration prior to beginning calculation of the public values in the usmDHKickstartTable.

If the CM configuration file contains SNMPv3 parameters, the CM MUST drop all SNMPv3 packets prior to calculating the public values in the usmDHKickstartTable.

8.5.2.2 SNMP Operation after CM Registration

After registration, the CM can be in one of the following SNMP operation modes:

- SNMPv1/v2c NmAccess mode
- SNMP Coexistence mode

Note: OpenAccess mode available in pre-3.0 DOCSIS OSSI specifications is not supported in DOCSIS 4.0.

The CM MUST NOT provide SNMP access if the configuration file does not contain SNMP access control TLVs such as docsDevNmAccessTable or SNMP coexistence TLV-11 or TLV-34, TLV-53 or TLV-54.

The SNMP mode of the CM is determined by the contents of the CM config file as follows:

- The CM is in SNMPv1/v2c NmAccess mode if the CM configuration file contains docsDevNmAccessTable settings for SNMP access control, does not contain SNMP coexistence TLV-11, TLV-34, TLV-38, TLV-53 or TLV-54 [MULPIv4.0].
- The CM is in SNMP coexistence mode if the CM configuration file contains snmpCommunityTable settings and/or TLV-34.1/34.2 and/or TLV-38. In this case, any entries made to the docsDevNmAccessTable are ignored.

8.5.2.3 SNMP NmAccessMode

SNMPv1/v2c NmAccess Mode (using docsDevNmAccess Table)

- The CM MUST implement docsDevNmAccessTable which controls access and trap destinations as described in [RFC 4639] for backward compatibility with pre-3.0 DOCSIS.
- The CM MUST process SNMPv1/v2c packets only in NmAccess mode and drop all SNMPv3 packets.

- The CM MUST NOT allow access to SNMPv3 MIBs as defined in [RFC 3411] through [RFC 3415] and [RFC 3584] while in NmAccess mode.

8.5.2.4 SNMP Coexistence Mode

The CM MUST process SNMPv1/v2c/v3 messages for SNMP Access Control and SNMP notifications as described by [RFC 3411] through [RFC 3415] and [RFC 3584] as follows:

- The SNMP-COMMUNITY-MIB controls the translation of SNMPv1/v2c packet community string into security name which select entries in the SNMP-USER-BASED-SM-MIB. Access control is provided by the SNMP-VIEW-BASED-ACM-MIB.
- SNMP-USER-BASED-SM-MIB and SNMP-VIEW-BASED-ACM-MIB controls SNMPv3 packets.
- Notification destinations are specified in the SNMP-TARGET-MIB, SNMP-NOTIFICATION-MIB and SNMP-VIEW-BASED-ACM-MIB.
- The CM MUST NOT provide access to docsDevNmAccessTable.
- When the CM is configured for SNMPv3, the CM MUST NOT allow SNMP access from the RF port during calculation of usmDHKickstartTable public value.
- When the CM is configured for SNMPv3, the CM MAY continue to allow access from the CPE port with the limited access as configured by the SNMP-COMMUNITY-MIB, SNMP-TARGET-MIB, SNMP-VIEW-BASED-ACM-MIB, and SNMP-USER-BASED-SM-MIB during calculation of usmDHKickstartTable public value.

8.5.2.5 SNMPv3 Initialization and Key Changes

Note that the SNMPv3 Initialization and Key Change process defined below is based on [RFC 2786], which always configures the SNMP agent with SNMPv3 HMAC-MD5-96 as the authentication protocol and CBC-DES as the privacy protocol, both specified in [RFC 3414]. Therefore, this specification does not provide a mechanism to initialize SNMPv3 using CFB128-AES-128 for privacy key, as defined in [RFC 3826] or any other configuration defined in [RFC 3414] and are left out of scope of this specification.

The DOCSIS 4.0 CM is designated as having a "very-secure" security posture in the context of [RFC 3414] and [RFC 3415], which means that default usmUserTable and VACM tables entries defined in Appendix A of [RFC 3414] and Appendix A of [RFC 3415] MUST NOT be present. The major implication for the CM is that only the config file can be used to provide the initial SNMPv3 security configuration.

[RFC 2786] provides a mechanism to kick start an SNMPv3 agent User-based Security Model [RFC 3414] and extensions to the same model for key change. [RFC 2786] does not define the mechanism to configure the initial key material for the kick start process. This specification defines the configuration requirements to initialize the SNMPv3 KickStart initialization defined in [RFC 2786] to configure SNMPv3 for the CM.

The CM MUST support the config file TLV-34 as defined in [MULPIv4.0] to configure the initial key material (KickStart Security Name and KickStart Public Number) used for the SNMPv3 agent initialization.

The TLV-34.1 KickStart Security Name corresponds to the SNMPv3 userName [RFC 3414] to be initialized in the CM.

The TLV-34.2 KickStart Public Number is a Diffie-Helman public number generated as described in the description of usmDHKickstartMgrPublic MIB object of [RFC 2786].

The CM MUST support a minimum of 5 entries of TLV-34 in the config file.

The CM MUST provide, by default, pre-defined entries in the USM table and VACM tables to correctly create the userName 'dhKickstart' with security level 'noAuthNoPriv' that has read-only access to system group and usmDHKickstartTable of [RFC 2786].

The CM MUST provide access to TLV-34 [MULPIv4.0] and dhKickstart defined userNames in usmUserTable as follows:

- Access as specified in the config file or the default access if corresponding to usernames defined above
- StorageType is 'permanent'
- Prohibit entry deletion
- Entries do not persist across MAC initialization

8.5.2.5.1 SNMPv3 Initialization

For each of up to five different TLV-34 (KickStart Security Name, KickStart Public Number) [MULPIv4.0] pairs from the configuration file, the CM MUST populate in the usmDHKickstartTable the MIB objects usmDHKickstartSecurityName and usmDHKickstartMgrPublic (each pair as an entry).

When a usmDHKickstartMgrPublic instance is set with a valid value during the initialization, the CM MUST create a corresponding row in the usmUserTable as defined in the clause description of usmDHKickstartMgrPublic MIB object of [RFC 2786].

After the CM has registered with the CMTS:

- The CM MUST populate the usmDHKickstartMyPublic MIB object of the usmDHKickstartTable as defined in [RFC 2786] for each entry that a non-zero length usmDHKickstartSecurityName and usmDHKickstartMgrPublic.
- [RFC 2786] Textual Convention DHKeyChange defines the mechanism to determine the Diffie-Helman shared secret for the CM and the SNMP manager. With the Diffie-Helman shared secret, the CM and other entities can derive the SNMPv3 privacy and authentication keys for the corresponding USM userName.
- The CM MUST derive the USM userName security and authentication keys as described in the description clause of the usmDHKickstartMgrPublic MIB object of [RFC 2786].

At this point the CM has completed its SNMPv3 initialization process.

After SNMPv3 initialization process has been finished, the CM MUST allow appropriate access level to a valid securityName with the correct authentication key and/or privacy key.

The CM MUST properly populate keys to appropriate tables as specified by the SNMPv3-related RFCs and [RFC 2786].

The following describes the process that the manager uses to derive the CM's unique authentication key and privacy key:

- The SNMP manager accesses the contents of the usmDHKickstartTable using the security name of 'dhKickstart' with no authentication.
- The SNMP manager gets the value of the CM's usmDHKickstartMyPublic number associated with the securityName for which the manager wants to derive authentication and privacy keys.
- Using the private random number, the manager can calculate the DH shared secret. From that shared secret, the manager can derive operational authentication and confidentiality keys for the securityName that the manager is going to use to communicate with the CM.

8.5.2.5.2 SNMPv3 initialization failure

In case of failure to complete SNMPv3 initialization (i.e., NMS cannot access CM via SNMPv3 PDU), the CM is in the SNMP Coexistence mode and will allow SNMPv1/v2c access if and only if the SNMP-COMMUNITY-MIB entries (and related entries) are configured.

8.5.2.5.3 DH Key Changes

The CMs MUST support the key-change mechanism specified in the textual convention DHKeyChange of [RFC 2786].

8.5.2.6 View-based Access Control Model (VACM) Profile

This section addresses the default VACM profile for DOCSIS CMs operating in SNMP Coexistence mode.

The CM MUST support pre-installed entries in VACM tables of [RFC 3415] as follows:

- The system manager, with full read/write/config access:
  vacmSecurityModel: 3 (USM)
  vacmSecurityName: docsisManager
  vacmGroupName: docsisManager
  vacmSecurityToGroupStorageType: permanent
  vacmSecurityToGroupStatus: active
- An operator/CSR with read/reset access to full modem:
  vacmSecurityModel: 3 (USM)
  vacmSecurityName: docsisOperator
  vacmGroupName: docsisOperator
  vacmSecurityToGroupStorageType: permanent
  vacmSecurityToGroupStatus: active
- RF Monitoring with read access to RF plant statistics:
  vacmSecurityModel: 3 (USM)
  vacmSecurityName: docsisMonitor
  vacmGroupName: docsisMonitor
  vacmSecurityToGroupStorageType: permanent
  vacmSecurityToGroupStatus: active
- User debugging with read access to 'useful' variables:
  vacmSecurityModel: 3 (USM)
  vacmSecurityName: docsisUser
  vacmGroupName: docsisUser
  vacmSecurityToGroupStorageType: permanent
  vacmSecurityToGroupStatus: active
- Group name to view translations
  vacmGroupName: docsisManager
  vacmAccessContextPrefix: "
  vacmAccessSecurityModel: 3 (USM)
  vacmAccessSecurityLevel: AuthPriv
  vacmAccessContextMatch: exact
  vacmAccessReadViewName: docsisManagerView
  vacmAccessWriteViewName: docsisManagerView
  vacmAccessNotifyViewName: docsisManagerView
  vacmAccessStorageType: permanent

vacmAccessStatus: active

vacmGroupName: docsisOperator

vacmAccessContextPrefix: "

vacmAccessSecurityModel: 3 (USM)

vacmAccessSecurityLevel: AuthNoPriv and AuthPriv

vacmAccessContextMatch: exact

vacmAccessReadViewName: docsisManagerView

vacmAccessWriteViewName: docsisOperatorWriteView

vacmAccessNotifyViewName: docsisManagerView

vacmAccessStorageType: permanent

vacmAccessStatus: active

vacmGroupName: docsisMonitor

vacmAccessContextPrefix: "

vacmAccessSecurityModel: 3 (USM)

vacmAccessSecurityLevel: AuthNoPriv and AuthPriv

vacmAccessContextMatch: exact

vacmAccessReadViewName: docsisMonitorView

vacmAccessWriteViewName: "

vacmAccessNotifyViewName: docsisMonitorView

vacmAccessStorageType: permanent

vacmAccessStatus: active

vacmGroupName: docsisUser

vacmAccessContextPrefix: "

vacmAccessSecurityModel: 3 (USM)

vacmAccessSecurityLevel: AuthNoPriv and AuthPriv

vacmAccessContextMatch: exact

vacmAccessReadViewName: docsisUserView

vacmAccessWriteViewName: "

vacmAccessNotifyViewName: "

vacmAccessStorageType: permanent

vacmAccessStatus: active

The CM includes, by default, the following views referred from the VACM entries above:

- docsisManagerView

  subtree: 1.3.6.1 (internet or entire MIB)
- docsisOperatorWriteView

  subtree: docsDevBase

  subtree: docsDevSoftware

object: docsDevEvControl

object: docsDevEvThrottleAdminStatus

- docsisMonitorView

  subtree: 1.3.6.1.2.1.1 (system)

  subtree: docsIfBaseObjects

  subtree: docsIfCmObjects

  docsisUserView.3.6.1.2.1.1 (system)

  subtree: docsDevBase

  object: docsDevSwOperStatus

  object: docsDevSwCurrentVers

  object:docsDevServerConfigFile

  subtree: docsDevEventTable

  subtree: docsDevCpeInetTable

  subtree: docsIfUpstreamChannelTable

  subtree: docsIfDownstreamChannelTable

  subtree: docsIfSignalQualityTable

- subtree

  subtree: 1

  docsIfCmStatusTable

The CM MUST also support additional VACM users as they are configured via an SNMP-embedded configuration file.

8.5.2.7 SNMPv1v2c Coexistence Configuration config file TLV

This section specifies CM processing requirements for the SNMPv1v2c Coexistence Configuration TLV [MULPIv4.0] when present in the configuration file. The SNMPv1v2c Coexistence Configuration TLV is used to configure SNMPv3 tables for SNMPv1 and v2c access. The CM MUST process SNMPv1v2c Coexistence Configuration TLV in conjunction with SNMP TLV-11 containing SNMPv3 tables, TLV-38, as well as SNMPv3 Access View Configuration TLV (see Section 8.5.2.8).

Based on the SNMPv1v2c Coexistence Configuration TLV, the CM MUST create entries in the following tables in order to cause the desired SNMP Access:

- snmpCommunityTable
- snmpTargetAddrTable
- vacmSecurityToGroupTable
- vacmAccessTable

The mapping from the TLV to these tables is described in the following section.

8.5.2.7.1 Mapping of TLV Fields into SNMPv3 Tables

The following section describes the mapping of SNMPv1v2c Coexistence Configuration TLV into SNMPv3 entries:

Table 31 - SNMPv1v2c Coexistence Configuration TLV Mapping

Sub-TLVs		Variable Name	Associated MIB Object
SNMPv1v2c Community Name		CommunityName	snmpCommunityName [RFC 3584]
SNMPv1v2c Transport Address Access			
	SNMPv1v2c Transport Address	TAddress	snmpTargetAddrTAddress [RFC 3413]
	SNMPv1v2c Transport Address Mask	TMask	snmpTargetAddrTMask [RFC 3584]
SNMPv1v2c Access View Type		AccessViewType	
SNMPv1v2c Access View Name		AccessViewName	vacmAccessReadViewName and vacmAccessWriteViewName [RFC 3415]

The CM is not required to verify the consistency of linkage of tables unless specified. It is intended that the SNMP agent will handle the corresponding configuration problems as part of the normal SNMP incoming requests (e.g., generating internal abstract data elements like noSuchView [RFC 3415]).

Table 33 through Table 38 describe the CM procedures to populate the SNMP Management Framework Message Processing and Access Control Subsystems [RFC 3412].

In configuring entries in these SNMPv3 tables, note the following:

- The ReadViewName and WriteViewName may correspond to default entries as defined in Section 8.5.2.5.2, individual entries defined by TLV-11 or entries created using SNMPv3 Access View Configuration (see Section 8.5.2.8).
- Several columnar objects are configured with indexes with the string "@CMconfig". If these tables are configured through other mechanisms, Network operators should not use values beginning with "@CMconfig" to avoid conflicts with the mapping process specified here.

8.5.2.7.2 snmpCommunityTable

The snmpCommunityTable is defined in the "SNMP Community MIB Module" section of [RFC 3584].

The CM MUST create one row in snmpCommunityTable for each SNMPv1v2c Coexistence Configuration TLV in the config file as follows:

- The CM MUST set in snmpCommunityIndex the keyword @CMconfig_n where 'n' is a sequential number starting at 0 for each TLV processed (e.g., "@CMconfig_0", "@CMconfig_1", etc.)
- The CM MUST create space separated tags in snmpCommunityTransportTag for each SNMPv1v2c Community Name sub-TLV of the SNMPv1v2c Coexistence Configuration TLV in the config file.

Table 32 - snmpCommunityTable

Column Name (* = Part of Index)	Column Value
* snmpCommunityIndex	"@CMconfig_n" where n is 0..m-1 and m is the number of SNMPv1v2c Community Name config file TLVs
snmpCommunityName	<CommunityName>
snmpCommunitySecurityName	"@CMconfig_n"
snmpCommunityContextEngineID	<the engineID of the cable modem>
snmpCommunityContextName	<Zero-length OCTET STRING>
snmpCommunityTransportTag	"@CMconfigTag_n" where n is 0..m-1 and m is the number of SNMPv1v2c Coexistence Configuration config file TLVs
snmpCommunityStorageType	volatile (2)
snmpCommunityStatus	active (1)

8.5.2.7.3 snmpTargetAddrTable

The snmpTargetAddrTable is defined in the "Definitions" section of [RFC 3413].

The CM MUST create one row in snmpTargetAddrTable for each SNMPv1v2c Transport Address Access sub-TLV of the SNMPv1v2c Coexistence Configuration TLV in the config file.

Table 33 - snmpTargetAddrTable

Column Name (* = Part of Index)	Column Value
* snmpTargetAddrName	"@CMconfigTag_n_i" where n is 0..m-1 and m is the number of SNMPv1v2c Coexistence Configuration config file TLVs. Where i is 0..p-1 and p is the number of SNMPv1v2c Transport Address Access sub-TLV within the SNMPv1v2c Coexistence Configuration config file TLV n
snmpTargetAddrTDomain	IPv4: snmpUDPDomain [RFC 3417] IPv6: transportDomainUdpIpv6 [RFC 3419]
snmpTargetAddrTAddress (IP Address and UDP Port)	IPv4: SnmpUDPAddress [RFC 3417] OCTET STRING (6) Octets 1-4: <TAddress> Octets 5-6: <TAddress> IPv6: TransportAddressIPv6 [RFC 3419] OCTET STRING (18) Octets 1-16: <TAddress> Octets 17-18: <TAddress>
snmpTargetAddrTimeout	Default from MIB
snmpTargetAddrRetryCount	Default from MIB
snmpTargetAddrTagList	"@CMconfigTag_n" where n is 0..m-1 and m is the number of SNMPv1v2c Coexistence Configuration config file TLVs
snmpTargetAddrParams	'00'h (null character)
snmpTargetAddrStorageType	volatile (2)
snmpTargetAddrRowStatus	active (1)

8.5.2.7.4 snmpTargetAddrExtTable

The snmpTargetAddrExtTable is defined in the "SNMP Community MIB Module" section of [RFC 3584].

The CM MUST create one row in snmpTargetAddrExtTable for each SNMPv1v2c Transport Address Access sub-TLV of the SNMPv1v2c Coexistence Configuration TLV in the config file.

Table 34 - snmpTargetAddrExtTable

Column Name (* = Part of Index)	Column Value
* snmpTargetAddrName	"@CMconfigTag_n_i" where n is 0..m-1 and m is the number of SNMPv1v2c Coexistence Configuration config file TLVs. Where i is 0..p-1 and p is the number of SNMPv1v2c Transport Address Access sub-TLV within the SNMPv1v2c Coexistence Configuration config file TLV n
snmpTargetAddrTMask	<Zero-length OCTET STRING> when <*TMask*> is not provided in the i^{th} SNMPv1v2c Transport Address Access sub-TLV IPv4: SnmpUDPAddress [RFC 3417] OCTET STRING (6) Octets 1-4: <*TMask*> Octets 5-6: <UDP Port> IPv6: TransportAddressIPv6 [RFC 3419] OCTET STRING (18) Octets 1-16: <*TMask*> Octets 17-18: <UDP Port>
snmpTargetAddrMMS	SM Maximum Message Size

8.5.2.7.5 vacmSecurityToGroupTable

The vacmSecurityToGroupTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create two rows in vacmSecurityGroupTable for each SNMPv1v2c Coexistence Configuration TLV in the config file as follows:

The CM MUST set in vacmSecurityName the keyword @CMconfig_n where 'n' is a sequential number starting at 0 for each SNMPv1v2c Coexistence Configuration TLV processed (e.g., "@CMconfig_0", "@CMconfig_1", etc.).

The CM MUST set in vacmGroupName the keyword @CMconfigV1_n for the first row and @CMconfigV2_n for the second row where 'n' is a sequential number starting at 0 for each SNMPv1v2c Coexistence Configuration TLV processed (e.g., "@CMconfigV1_0", "@CMconfigV1_1", etc.).

Table 35 - vacmSecurityToGroupTable

Column Name (* = Part of Index)	First Row Column Value	Second Row Column Value
* vacmSecurityModel	SNMPV1 (1)	SNMPV2c (2)
* vacmSecurityName	"@CMconfig_n"	"@CMconfig_n"
vacmGroupName	"@CMconfigV1_n"	"@CMconfigV2_n"
vacmSecurityToGroupStorageType	volatile (2)	volatile (2)
vacmSecurityToGroupStatus	active (1)	active (1)

8.5.2.7.6 vacmAccessTable

The vacmAccessTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create two rows in vacmAccessTable for each SNMPv1v2c Coexistence Configuration TLV in the config file as follows:

> The CM MUST set in vacmGroupName the keyword @CMconfigV1_n for the first row and @CMconfigV2_n for the second row where 'n' is a sequential number starting at 0 for each SNMPv1v2c Coexistence Configuration TLV processed (e.g., "@CMconfigV1_0", "@CMconfigV1_1", etc.).

Table 36 - vacmAccessTable

Column Name (* = Part of Index)	Column Value	Column Value
* vacmGroupName	"@CMconfigV1_n"	"@CMconfigV2_n"
* vacmAccessContextPrefix	<zero-length string>	<zero-length string>
* vacmAccessSecurityModel	SNMPV1 (1)	SNMPV2c (2)
* vacmAccessSecurityLevel	noAuthNoPriv (1)	noAuthNoPriv (1)
vacmAccessContextMatch	exact (1)	exact (1)
vacmAccessReadViewName	Set <AccessViewName>	Set <AccessViewName>
vacmAccessWriteViewName	When <AccessViewType> == '2' Set <AccessViewName> Otherwise, set <Zero-length OCTET STRING>	When <AccessViewType> == '2' Set <AccessViewName> Otherwise, set <Zero-length OCTET STRING>
vacmAccessNotifyViewName	<Zero-length OCTET STRING>	<Zero-length OCTET STRING>
vacmAccessStorageType	volatile (2)	volatile (2)
vacmAccessStatus	active (1)	active (1)

8.5.2.8 SNMPv3 Access View Configuration config file TLV

This section specifies CM processing requirements for SNMPv3 Access View Configuration TLVs when present in the configuration file. The SNMPv3 Access View Configuration TLV is used to configure the table vacmViewTreeFamilyTable in a simplified way. The CM MUST process SNMPv3 Access View Configuration TLV in conjunction with SNMP TLV-11 containing SNMPv3 tables, TLV-38 as well as SNMPv1v2c Coexistence Configuration TLV (see Section 8.5.2.7).

The mapping from the TLV to these tables is described in the following section.

8.5.2.8.1 Mapping of TLV Fields into SNMPv3 Tables

The following section describes the mapping of SNMPv3 Access View Configuration TLVs into vacmViewTreeFamilyTable:

Table 37 - SNMPv3 Access View Configuration TLV Mapping

Sub-TLVs	Variable Name	Associated MIB Object [RFC 3415]
SNMPv3 Access View Name	AccessViewName	vacmViewTreeFamilyViewName
SNMPv3 Access View Subtree	AccessViewSubTree	vacmViewTreeFamilySubtree

Sub-TLVs	Variable Name	Associated MIB Object [RFC 3415]
SNMPv3 Access View Mask	AccessViewMask	vacmViewTreeFamilyMask
SNMPv3 Access View Type	AccessViewType	vacmViewTreeFamilyType

Disconnected entries in the CM SNMP access configuration database are not expected to be detected by the CM as part of the configuration. Eventually, the SNMP agent will not grant access to SNMP requests, for example, to disconnected Security Names and View trees as a result of a TLV configuration mistake.

Table 38 describes the CM procedures to populate the SNMP Management Framework Access Control Subsystem [RFC 3412].

In configuring entries for SNMPv3 Access View Configuration TLV, note the following:

> One entry is created for each TLV. Some Access Views may have a number of included/excluded OID branches. Only Access View Name will be common for all these OID branches. To support such type of Access View with multiple included/excluded OID branches a number of multiple SNMPv3 Access View Configuration TLVs need to be defined in configuration file.

8.5.2.8.2 vacmViewTreeFamilyTable

The vacmViewTreeFamilyTable is defined in the "Definitions" section of [RFC 3415].

The CM MUST create one row in vacmViewTreeFamilyTable for each SNMPv3 Access View Configuration TLV in the config file. The CM MUST reject the config file if two SNMPv3 Access View Configuration TLVs have identical index components relative to vacmViewTreeFamilyTable. In such instance, the CM would not be able to create an entry for the second TLV containing the duplicate index.

The CM MUST set the object vacmViewTreeFamilySubtree to 1.3.6 when no sub-TLV SNMPv3 Access View Subtree is defined in the config file.

The CM MUST set the object vacmViewTreeFamilyMask to the default zero-length string when no sub-TLV SNMPv3 Access View Mask is defined.

The CM MUST set the object vacmViewTreeFamilyType to the default value 1 (included) when no sub-TLV SNMPv3 Access View Type is defined.

Table 38 - vacmViewTreeFamilyTable

Column Name (* = Part of Index)	Column Value
* vacmViewTreeFamilyViewName	<AccessViewName>
* vacmViewTreeFamilySubtree	<AccessViewSubTree>
vacmViewTreeFamilyMask	<AccessViewMask>
vacmViewTreeFamilyType	<AccessViewType>
vacmViewTreeFamilyStorageType	volatile (2)
vacmViewTreeFamilyStatus	active (1)

8.5.2.9 SNMP CPE Access Control Configuration config file TLV

The 'SNMP CPE Access Control' config File TLV (see [MULPIv4.0]) provides a mechanism to filter SNMP PDU-requests originating from a CMCI interface.

The CM MUST enforce the requirements of 'SNMP CPE Access Control' when configured in SNMP Coexistence mode.

The CM MAY ignore the 'SNMP CPE Access Control' encodings when configured in NmAccess mode.

When applicable, the CM MUST enforce the 'SNMP CPE Access Control' requirements to enable or disable SNMP Access originating from a CMCI interface directed to any CM provisioned IP addresses (see [MULPIv4.0]) or any of the CM's CMCI IP addresses defined in Section 9.1, and prior to SNMP protocol specific access control mechanisms such as SNMPv3 Access View, or NmAccess settings.

8.5.3 Security Management UML Information Model

This section defines the UML Information Model for CM security management functions.

The CM objects for Security Management are shown in Figure 8.

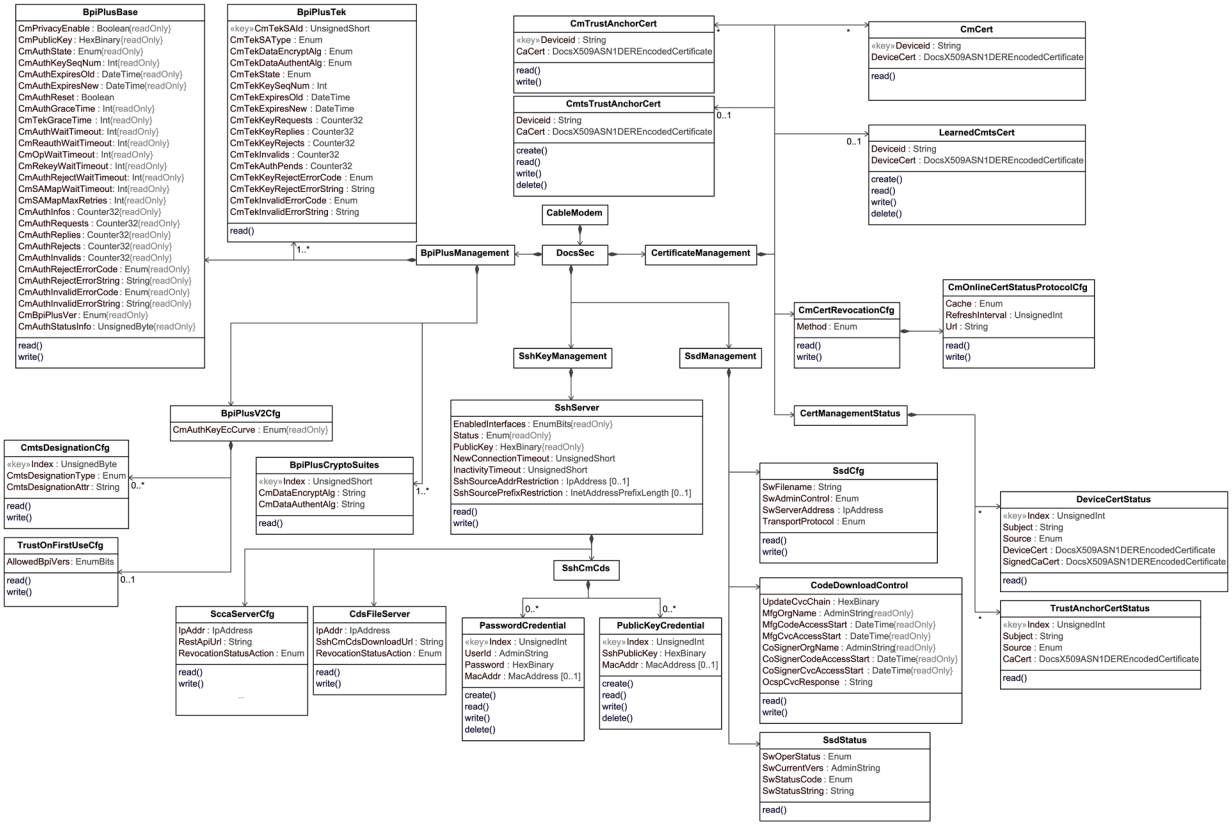


Figure 8 - CM Security Management Information Model

8.5.3.1 DocsSec

The DocsSec object serves as the root of the CM Security Management Information Model.

Table 39 - DocsSec Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
BpiPlusManagement	Directed composition to BpiPlusManagement	1	1	
CertificateManagement	Directed composition to CertificateManagement	1	1	
SsdManagement	Directed composition to SsdManagement	1	1	
SshKeyManagement	Directed composition to SshKeyManagement	1	1	

8.5.3.2 BpiPlusManagement

The BpiPlusManagement object describes the control of the BPI+V1 and V2 functions.

Reference: [RFC 4131]

Table 40 - BpiPlusManagement Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
BpiPlusBase	Directed composition to BpiPlusBase	1	1	
BpiPlusTek	Directed composition to BpiPlusTek	1	1..*	
BpiPlusV2Cfg	Directed composition to BpiPlusV2Cfg	1	1	
BpiPlusCryptoSuites	Directed composition to BpiPlusCryptoSuites	1	1..*	

8.5.3.3 BpiPlusBase

The BpiPlusBase object describes the basic and authorization-related BPI+ attributes.

Reference: docsBpi2CmBaseTable [RFC 4131]

Table 41 - BpiPlusBase Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
CmPrivacyEnable	Boolean	R/O		N/A	
CmPublicKey	HexBinary	R/O	SIZE (0..524)	N/A	
CmAuthState	Enum	R/O	start(1), authWait(2), authorized(3), reauthWait(4), authRejectWait(5), silent(6)	N/A	
CmAuthKeySeqNum	Int	R/O	0..15	N/A	
CmAuthExpiresOld	DateTime	R/O	SIZE(11)	N/A	
CmAuthExpiresNew	DateTime	R/O	SIZE(11)	N/A	
CmAuthReset	Boolean	R/W		N/A	
CmAuthGraceTime	Int	R/O	1..6047999	seconds	
CmTekGraceTime	Int	R/O	1..302399	seconds	
CmAuthWaitTimeout	Int	R/O	1..30	seconds	
CmReauthWaitTimeout	Int	R/O	1..30	seconds	
CmOpWaitTimeout	Int	R/O	1..10	seconds	
CmRekeyWaitTimeout	Int	R/O	1..10	seconds	
CmAuthRejectWaitTimeout	Int	R/O	1..600	seconds	
CmSAMapWaitTimeout	Int	R/O	1..10	seconds	
CmSAMapMaxRetries	Int	R/O	1..10	retries	
CmAuthInfos	Counter32	R/O		N/A	
CmAuthRequests	Counter32	R/O		N/A	
CmAuthReplies	Counter32	R/O		N/A	
CmAuthRejects	Counter32	R/O		N/A	
CmAuthInvalids	Counter32	R/O		N/A	
CmAuthRejectErrorCode	Enum	R/O	none(1), unknown(2), unauthorizedCm(3), unauthorizedSaid(4), permanentAuthorizationFailure(8), timeOfDayNotAcquired(11) eaeDisabled(12) unsupportedBpiVer(13)	N/A	

Attribute Name	Type	Access	Type Constraints	Units	Default Value
CmAuthRejectErrorString	String	R/O	SIZE (0..128)	N/A	
CmAuthInvalidErrorCode	Enum	R/O	none(1), unknown(2), unauthorizedCm(3), unsolicited(5), invalidKeySequence(6), keyRequestAuthenticationFailure(7)	N/A	
CmAuthInvalidErrorString	String	R/O	SIZE (0..128)	N/A	
CmBpiPlusVer	Enum	R/O	none(1), v1(2), v2(3)	N/A	
CmAuthStatusInfo	UnsignedByte	R/O		N/A	

8.5.3.3.1 CmPrivacyEnable

This attribute identifies whether this CM is provisioned to run Baseline Privacy Plus.

Reference: docsBpi2CmPrivacyEnable [RFC 4131]

8.5.3.3.2 CmPublicKey

The value of this attribute is a DER-encoded RSAPublicKey ASN.1 type string, as defined in [X.509], corresponding to the public key of the CM. This is the zero-length value if the CMTS does not retain the public key. This attribute is only used in BPI+V1 mode.

Reference: docsBpi2CmPublicKey [RFC 4131]

8.5.3.3.3 CmAuthState

The value of this attribute is the state of the CM authorization FSM. The start state indicates that FSM is in its initial state. The possible values for this attribute are listed below:

'start' - This is the initial state of the FSM. No resources are assigned to or used by the FSM, all timers are off, and no processing is scheduled.

'authWait' - This is the state of the FSM when the CM has sent both an Authentication Information and an Authorize Request message and is waiting for the reply.

'authorized' - This is the state of the FSM when the CM has received an Authorization Reply message and has a valid Authorization Key and the list of SAIDs. Transition into this state triggers the creation of one TEK FSM for each of the CM's privacy-enabled SAIDs.

'reauthWait' - This is the state of the FSM when the CM has sent an outstanding reauthorization request to CMTS and is waiting for a response.

'authRejectWait' - This is the state of the FSM when the CM has received an Authorization Reject message in response to its last Authorization Request and the error code indicated that the error was NOT permanent or that EAE is not disabled.

'silent' - This is the state of the FSM when the CM has received an Authorization Reject message in response to its last Authorization Request and the error code indicated that the error was permanent (e.g., Device Certificate is revoked) or the CM and CMTS fail to properly authenticate.

Reference: docsBpi2CmAuthState [RFC 4131]

8.5.3.3.4 CmAuthKeySeqNum

The value of this attribute is the most recent authorization key sequence number for the CM authorization FSM.

Reference: docsBpi2CmAuthKeySequenceNumber [RFC 4131]

8.5.3.3.5 CmAuthExpiresOld

The value of this attribute is the actual clock time for expiration of the immediate predecessor of the most recent authorization key for the CM authorization FSM. If this FSM has only one authorization key, then the value is the time of activation of this FSM.

Reference: docsBpi2CmAuthExpiresOld [RFC 4131]

8.5.3.3.6 CmAuthExpiresNew

The value of this attribute is the actual clock time for expiration of the most recent authorization key for this FSM.

Reference: docsBpi2CmAuthExpiresNew [RFC 4131]

8.5.3.3.7 CmAuthReset

When the value of this attribute is 'true', the CM, generates a Reauthorize event in the CM authorization FSM.

Reference: docsBpi2CmAuthReset [RFC 4131]

8.5.3.3.8 CmAuthGraceTime

The value of this attribute is the grace time for an authorization key in seconds. A CM is expected to start trying to get a new authorization key beginning AuthGraceTime seconds before the most recent authorization key actually expires.

Reference: docsBpi2CmAuthGraceTime [RFC 4131]

8.5.3.3.9 CmTekGraceTime

The value of this attribute is the grace time for the TEK in seconds. The CM is expected to start trying to acquire a new TEK beginning TEK GraceTime seconds before the expiration of the most recent TEK.

Reference: docsBpi2CmTEKGraceTime [RFC 4131]

8.5.3.3.10 CmAuthWaitTimeout

The value of this attribute is the Authorize Wait Timeout in seconds.

Reference: docsBpi2CmAuthWaitTimeout [RFC 4131]

8.5.3.3.11 CmReauthWaitTimeout

The value of this attribute is the Reauthorize Wait Timeout in seconds.

Reference: docsBpi2CmReauthWaitTimeout [RFC 4131]

8.5.3.3.12 CmOpWaitTimeout

The value of this attribute is the Operational Wait Timeout in seconds.

Reference: docsBpi2CmOpWaitTimeout [RFC 4131]

8.5.3.3.13 CmReKeyWaitTimeout

The value of this attribute is the Rekey Wait Timeout in seconds.

Reference: docsBpi2CmRekeyWaitTimeout [RFC 4131]

8.5.3.3.14 CmAuthRejectWaitTimeout

The value of this attribute is the Authorization Reject Wait Timeout in seconds.

Reference: docsBpi2CmAuthRejectWaitTimeout [RFC 4131]

8.5.3.3.15 CmSAMapWaitTimeout

The value of this attribute is retransmission interval, in seconds, of SA Map Requests from the MAP Wait state.

Reference: docsBpi2CmSAMapWaitTimeout [RFC 4131]

8.5.3.3.16 CmSAMapMaxRetries

The value of this attribute is the maximum number of Map Request retries allowed.

Reference: docsBpi2CmSAMapMaxRetries [RFC 4131]

8.5.3.3.17 CmAuthInfos

The value of this attribute is the number of times the CM has transmitted an Authentication Information message. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmAuthentInfos [RFC 4131]

8.5.3.3.18 CmAuthRequests

The value of this attribute is the number of times the CM has transmitted an Authorization Request message. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmAuthRequests [RFC 4131]

8.5.3.3.19 CmAuthReplies

The value of this attribute is the number of times the CM has transmitted an Authentication Reply message. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmAuthReplies [RFC 4131]

8.5.3.3.20 CmAuthRejects

The value of this attribute is the number of times the CM has transmitted an Authentication Reject message. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmAuthRejects [RFC 4131]

8.5.3.3.21 CmAuthInvalids

The value of this attribute is the count of times the CM has received an Authorization Invalid message. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmAuthInvalids [RFC 4131]

8.5.3.3.22 CmAuthRejectErrorCode

The value of this attribute is the enumerated description of the Error-Code in the most recent Authorization Reject message received by the CM. The possible values for this attribute are listed below:

'none' - No Authorization Reject message has been received since reboot.

'unknown' - The last Error-Code value was 0 (i.e., no information).

'unauthorizedCm' - The CM is unauthorized.

'unauthorizedSaid' - The SAID is unauthorized.

'permanentAuthorizationFailure' - The error conditions affecting the BPKM authorization exchange. These include the following:

- an unknown manufacturer, i.e., the CMTS does not have the CA certificate belonging to the issuer of a CM Device Certificate;
- CM Device Certificate has an invalid signature;
- ASN.1 parsing failure during verification of CM Device Certificate;
- CM Device Certificate is revoked (see [SECv4.0] Certificate Revocation section);
- inconsistencies between certificate data and data in accompanying BPKM attributes; and
- the CM and CMTS have incompatible security capabilities.

'timeOfDayNotAcquired' - Time of day is not acquired.

'eaeDisabled' - EAE is disabled.

'unsupportedBpiVer' - The BPI+ version used by CM is not supported.

Reference: docsBpi2CmAuthRejectErrorCode [RFC 4131]; Section 7.1.3.16, "Requirements for DOCSIS Baseline Privacy Plus MIB (RFC 4131)"

8.5.3.3.23 CmAuthRejectErrorString

The value of this attribute is text string in the most recent Authorization Reject message received by the CM. This is a zero length string if no Authorization Reject message has been received since reboot.

Reference: docsBpi2CmAuthRejectErrorString [RFC 4131]

8.5.3.3.24 CmAuthInvalidErrorCode

The value of this attribute is the enumerated description of the Error-Code in the most recent Authorization Invalid message received by the CM. The possible values for this attribute are listed below:

'none' - No Authorization Invalid message has been received since reboot.

'unknown' - The last Error-Code value was 0 (i.e., no information).

'unauthorizedCm' - The CM is unauthorized.

'unsolicited' - The CMTS sends an unsolicited Authorization Invalid message to a CM, forcing an Auth Invalid event.

'invalidKeySequence' - The Key Sequence Number is invalid.

'keyRequestAuthenticationFailure' - The Key Request Message fails authentication with the CMTS.

Reference: docsBpi2CmAuthInvalidErrorCode [RFC 4131]

8.5.3.3.25 CmAuthInvalidErrorString

The value of this attribute is text string in the most recent Authorization Invalid message received by the CM. This is a zero-length string if no Authorization Invalid message has been received since reboot.

Reference: docsBpi2CmAuthInvalidErrorString [RFC 4131]

8.5.3.3.26 CmBpiPlusVer

The value of this attribute shows the BPI+ version used by the CM for authentication.

'none' - BPI+ is not used by the CM.

'v1' - BPI+V1

'v2' - BPI+V2

Reference: Baseline Privacy Key Management (BPKM) Protocol [SECv4.0]

8.5.3.3.27 CmAuthStatusInfo

The value of this attribute is the number of times the CM has transmitted an Authentication Status Info message. This attribute is for BPI+V2 mode only.

For a non-BPI+V2 CM, the value of this attribute needs to be '0'. If this is a BPI+V2 CM, this attribute needs to be a non-zero value.

The value of this attribute is not preserved after the CM power-cycles or resets.

Reference: Baseline Privacy Key Management (BPKM) Protocol [SECv4.0]

8.5.3.4 BpiPlusTek

The BpiPlusTek object describes the attributes of each CM Traffic Encryption Key (TEK) association. The CM maintains (no more than) one TEK association per SAID.

Reference: docsBpi2CmTEKTable [RFC 4131]

Table 42 - BpiPlusTek Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
CmTekSAId	UnsignedShort	Key	1..16383	N/A	
CmTekSAType	Enum	R/O	none(1), primary(2), static(3), dynamic(4)	N/A	
CmTekDataEncryptAlg	Enum	R/O	none(1), des56CbcMode(2), des40CbcMode(3), aes128CbcMode(4), aes256CbcMode(5)	N/A	
CmTekDataAuthentAlg	Enum	R/O	none(1)	N/A	
CmTekState	Enum	R/O	start(1), opWait(2), opReauthWait(3), operational(4), rekeyWait(5), rekeyReauthWait(6)	N/A	
CmTekKeySeqNum	Int	R/O	0..15	N/A	
CmTekExpiresOld	DateTime	R/O	SIZE(11)	N/A	
CmTekExpiresNew	DateTime	R/O	SIZE(11)	N/A	
CmTekKeyRequests	Counter32	R/O		N/A	
CmTekKeyReplies	Counter32	R/O		N/A	
CmTekKeyRejects	Counter32	R/O		N/A	
CmTekInvalids	Counter32	R/O		N/A	
CmTekAuthPends	Counter32	R/O		N/A	
CmTekKeyRejectErrorCode	Enum	R/O	none(1), unknown(2), unauthorizedSaid(4)	N/A	
CmTekKeyRejectErrorString	String	R/O	SIZE (0..128)	N/A	
CmTekInvalidErrorCode	Enum	R/O	none(1), unknown(2), invalidKeySequence(6)	N/A	
CmTekInvalidErrorString	String	R/O	SIZE (0..128)	N/A	

8.5.3.4.1 CmTekSAId

The value of this key attribute is the DOCSIS Security Association ID (SAID).

Reference: docsBpi2CmTEKSAId [RFC 4131]

8.5.3.4.2 CmTekSAType

The value of this attribute is the type of security association.

'none' - SA is empty.

'primary' - Primary SA

'static' - Static SA

'dynamic' - Dynamic SA

Reference: docsBpi2CmTEKSAType [RFC 4131]

8.5.3.4.3 CmTekDataEncryptAlg

The value of this attribute is the data encryption algorithm for this SAID.

'none' - No data encryption algorithm is used.

'des56CbcMode' - CBC-Mode, 56-bit DES

'des40CbcMode' - CBC-Mode, 40-bit DES

'aes128CbcMode' - CBC-Mode, 128-bit block, 128-bit key AES

'aes256CbcMode' - CBC-Mode, 128-bit block, 256-bit key AES

Reference: docsBpi2CmTEKDataEncryptAlg [RFC 4131]

8.5.3.4.4 CmTekDataAuthentAlg

The value of this attribute is the data authentication algorithm for this SAID.

'none' - No data authentiction algorithm is used.

Reference: docsBpi2CmTEKDataAuthentAlg [RFC 4131]

8.5.3.4.5 CmTekState

The value of this attribute is the state of the indicated TEK FSM. The start(1) state indicates that the CM TEK FSM is in its initial state. The possible values for this attribute are listed below:

'start' - This is the initial state of the FSM. No resources are assigned to or used by the FSM, all timers are off, and no processing is scheduled.

'opWait' - The TEK state machine has sent its initial Key Request for its SAID's keying material (TEK and CBC IV) and is waiting for a reply from the CMTS.

'opReauthWait' - The Authorization state machine is in a reauthorization cycle, and the CM does not have valid keying material for this SAID.

'operational' - The CM has valid keying material for the associated SAID.

'rekeyWait' - The TEK Refresh Timer has expired, and the CM has requested a key update for this SAID to replace the older of the two TEKs.

'rekeyReauthWait' - The CM has valid traffic keying material for this SAID and has an outstanding request for the latest keying material, and the Authorization state machine has initiated a reauthorization cycle.

Reference: docsBpi2CmTEKState [RFC 4131]

8.5.3.4.6 CmTekKeySeqNum

The value of this attribute is the most recent TEK key sequence number for this TEK FSM.

Reference: docsBpi2CmTEKKeySequenceNumber [RFC 4131]

8.5.3.4.7 CmTekExpiresOld

The value of this attribute is actual clock time for expiration of the immediate predecessor of the most recent TEK for this FSM. If this FSM has only one TEK, then the value is the time of activation of this FSM.

Reference: docsBpi2CmTEKExpiresOld [RFC 4131]

8.5.3.4.8 CmTekExpiresNew

The value of this attribute is the actual clock time for expiration of the most recent TEK for this FSM.

Reference: docsBpi2CmTEKExpiresNew [RFC 4131]

8.5.3.4.9 CmTekKeyRequests

The value of this attribute is the number of times the CM has transmitted a Key Request message. Discontinuities in the value of this counter can occur at re-initialization of the management system.

Reference: docsBpi2CmTEKKeyRequests [RFC 4131]

8.5.3.4.10 CmTekKeyReplies

The value of this attribute is the number of times the CM has received a Key Reply message, including a message whose authentication failed. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmTEKKeyReplies [RFC 4131]

8.5.3.4.11 CmTekKeyRejects

The value of this attribute is the number of times the CM has received a Key Reject message, including a message whose authentication failed. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmTEKKeyRejects [RFC 4131]

8.5.3.4.12 CmTekKeyInvalids

The value of this attribute is the number of times the CM has received a Key Reply message, including a message whose authentication failed. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ifCounterDiscontinuityTime.

Reference: docsBpi2CmTEKInvalids [RFC 4131]

8.5.3.4.13 CmTekAuthPends

The value of this attribute is the count of times an Authorization Pending (Auth Pend) event occurred in the TEK FSM. Discontinuities in the value of this counter can occur at re-initialization of the management system.

Reference: docsBpi2CmTEKAuthPends [RFC 4131]

8.5.3.4.14 CmTekKeyRejectErrorCode

The value of this attribute is the enumerated description of the Error-Code in the most recent Key Reject message received by the CM. This has the value unknown(2) if the last Error-Code value was 0 and none(1) if no Key Reject message has been received since registration.

'none' - No key reject message has been received since registration.

'unknown' - Last error code value was zero.

'unauthorizedSaid' - SA ID is unauthorized.

Reference: docsBpi2CmTEKKeyRejectErrorCode [RFC 4131]

8.5.3.4.15 CmTekKeyRejectErrorString

The value of this attribute is the text string in the most recent Key Reject message received by the CM. This is a zero length string if no Key Reject message has been received since registration.

Reference: docsBpi2CmTEKKeyRejectErrorString [RFC 4131]

8.5.3.4.16 CmTekKeyInvalidErrorCode

The value of this attribute is the enumerated description of the Error-Code in the most recent TEK Invalid message received by the CM. This has the value unknown(2) if the last Error-Code value was 0 and none(1) if no TEK Invalid message has been received since registration.

'none' - No key invalid message has been received since registration.

'unknown' - Last error code value was zero.

'invalidKeySequence' - Key sequence number is invalid.

Reference: docsBpi2CmTEKInvalidErrorCode [RFC 4131]

8.5.3.4.17 CmTekKeyInvalidErrorString

The value of this attribute is the text string in the most recent TEK Invalid message received by the CM. This is a zero length string if no Key Reject message has been received since registration.

Reference: docsBpi2CmTEKInvalidErrorString [RFC 4131]

8.5.3.5 BpiPlusCryptoSuites

The BpiPlusCryptoSuites object describes the Baseline Privacy Plus cryptographic suite capabilities.

Reference: docsBpi2CmCryptoSuiteTable [RFC 4131]

Table 43 - BpiPlusCryptoSuites Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Index	UnsignedShort	Key	1..1000	N/A	
CmDataEncryptAlg	Enum	R/O	reserved(1), des56CbcMode(2), des40CbcMode(3), aes128CbcMode(4), aes256CbcMode(5)		
CmDataAuthentAlg	Enum	R/O	none(1), reserved(2)		

8.5.3.5.1 Index

The value of this attribute is the index for a cryptographic suite.

Reference: docsBpi2CmCryptoSuiteIndex [RFC 4131]

8.5.3.5.1 CmDataEncryptAlg

The value of this attribute is the data encryption algorithm for this cryptographic suite capability.

'reserved' -This value is reserved for further use.

'des56CbcMode' - CBC-Mode, 56-bit DES

'des40CbcMode' - CBC-Mode, 40-bit DES

'aes128CbcMode' - CBC-Mode, 128-bit block, 128-bit key AES

'aes256CbcMode' - CBC-Mode, 128-bit block, 256-bit key AES

Reference: docsBpi2CmCryptoSuiteDataEncryptAlg [RFC 4131]

8.5.3.5.1 CmDataAuthentAlg

The value of this attribute is the data authentication algorithm for this cryptographic suite capability.

'none' - No data authentiction algorithm is used.

'reserved' -This value is reserved for further use.

Reference: docsBpi2CmCryptoSuiteDataAuthentAlg [RFC 4131]

8.5.3.6 BpiPlusV2Cfg

TheBpiPlusV2Cfg object describes the control of BPI+V2 on the CM.

Reference: Baseline Privacy Key Management (BPKM) Protocol [SECv4.0]

Table 44 - BpiPlusV2Cfg Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
CmtsDesignationCfg	Directed composition to CmtsDesignationCfg	1	0..*	
TrustOnFirstUseCfg	Directed composition to TrustOnFirstUseCfg	1	0..1	

Table 45 - BpiPlusV2Cfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
CmAuthKeyEcCurve	Enum	R/O	secp256r1(1), secp384r1(2), secp521r1(3), x25519(4), x448(5)		

8.5.3.6.1 CmAuthKeyEcCurve

The value of this attribute is the supported Elliptic-Curve Groups (ECDHE) for CM to generate an EC Auth Key while in BPI+V2 mode.

'secp256r1' - also known as NIST P-256 curve

'secp384r1' - also known as NIST P-384 curve

'secp521r1'- also known as NIST P-521 curve

'x25519' - curve25519 with Diffie-Hellman key agreement

'x448' - curve448 with Diffie-Hellman key agreement

Reference: Baseline Privacy Key Management (BPKM) Protocol [SECv4.0]

8.5.3.7 CmtsDesignationCfg

The CmtsDesignationCfg object contains information the CM uses to provide rules to bind a CM to a CMTS. Multiple rules can be added to provide the level of granularity desired.

Reference: CMTS-Designation [SECv4.0]

Table 46 - CmtsDesignationCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Index	UnsignedByte	Key		N/A	
CmtsDesignationType	Enum	R/W	fingerprint(1), cn(2), ou(3), on(4), serialNum(5), caFingerprint(6), caCN(7), caOU(8), caON(9), caSerialNum(10),	N/A	
CmtsDesignationAttr	String	R/W	SIZE (0..128)	N/A	

8.5.3.7.1 Index

The value of this key attribute is the index for an instance of this object.

8.5.3.7.2 CmtsDesignationType

The value of this attribute indicates the type of data in the certificates to be checked for CM at the next authorization session. The possible values for this attribute are listed below:

'fingerprint' - this data type is the fingerprint of CMTS Device Certificate.

'cn' - this data type is the Common Name of CMTS Device Certificate.

'ou' - this data type is the Organizational Unit of CMTS Device Certificate.

'on' - this data type is the Organization Name of CMTS Device Certificate.

'serialNum' - this data type is the serial number of CMTS Device Certificate.

'caFingerprint' - this data type is the fingerprint of CMTS Device CA Certificate.

'caCN' - this data type is the Common Name of CMTS Device CA Certificate.

'caOU' - this data type is the Organizational Unit of CMTS Device CA Certificate.

'caON' - this data type is the Organization Name of CMTS Device CA Certificate.

'caSerialNum' - this data type is the serial number of CMTS Device CA Certificate.

Reference: CMTS-Designation [SECv4.0]

8.5.3.7.3 CmtsDesignationAttr

The value of this attribute is used when checking the value of specified data type for CM at the next authorization session.

Reference: CMTS-Designation [SECv4.0]

8.5.3.8 TrustOnFirstUseCfg

The TrustOnFirstUseCfg object contains information for CM to prevent the unauthorized downgrades. This object records the successful authentication configuration used during the first connection to a CMTS and is checked and used in subsequent connections.

Reference: Trust On First Use (TOFU) [SECv4.0]

Table 47 - TrustOnFirstUseCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
AllowedBpiVers	EnumBits	R/W	bpiPlusV1(0), bpiPlusV2(1)	N/A	

8.5.3.8.1 AllowedBpiVers

This attribute configures which BPI+ version the CM supported in subsequent connections.

If the 'bpiPlusV1' bit (bit 0) is set to '1', the CM only uses BPI+V1 to connect to a CMTS.

If the 'bpiPlusV2' bit (bit 1) is set to '1', the CM only uses BPI+V2 to connect to a CMTS.

If both Bit 0 and Bit 1 are set, the CM allows access from both BPI+V1 and BPI+V2 to connect to a CMTS.

Reference: Trust On First Use (TOFU) [SECv4.0]

8.5.3.9 CertificateManagement

The CertificateManagement object describes the set of known Certificate Authority certificates, Device certificates as well as revocation control and certs status acquired by the CM.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

Table 48 - CertificateManagement Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
CmTrustAnchorCert	Directed composition to TrustAnchorCert	1	*	
CmtsTrustAnchorCert	Directed composition to CmtsTrustAnchorCert	1	0..1	
CmCert	Directed composition to CmCert	1	*	
LearnedCmtsCert	Directed composition to LearnedCmtsCert	1	0..1	
CertRevocationCfg	Directed composition to CertRevocationCfg	1	1	
CertManagementStatus	Directed composition to CertManagementStatus	1	1	

8.5.3.10 CmTrustAnchorCert

The CmTrustAnchorCert object describes the set of known CM CA certificates: the Device CA certificates from the new PKI, and the Manufacturer CA certificate from the legacy PKI.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0]

Table 49 - CmTrustAnchorCert Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
DeviceId	String	Key		N/A
CaCert	DocsX509ASN1DEREncodedCertificate	R/O		N/A

8.5.3.10.1 DeviceId

This key attribute represents the unique identifier of an instance of this object.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0]

8.5.3.10.2 CaCert

This attribute represents the X509 DER-encoded CA certificate that signed the CM device certificate.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0],
docsBpi2CmDeviceManufCert [RFC 4131] (Legacy PKI CM certificate functions),
docsBpi2Ext31CmDeviceManufCert [DOCS-BPI2EXT-MIB] (DOCSIS 3.1/4.0 PKI CM certificate functions)

8.5.3.11 CmtsTrustAnchorCert

The CmtsTrustAnchorCert object describes the set of learned CMTS CA certificates during the BPI+V2 process.

Reference: Baseline Privacy Key Management (BPKM) protocol [SECv4.0]

Table 50 - CmtsTrustAnchorCert Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
DeviceId	String	R/W		N/A
CaCert	DocsX509ASN1DEREncodedCertificate	R/W		N/A

8.5.3.11.1 DeviceId

This attribute represents the unique identifier of an instance of this object.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0]

8.5.3.11.2 CaCert

This attribute represents the X509 DER-encoded CA certificate that signed the CMTS device certificate.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0]

8.5.3.12 CmCert

The CmCert object describes the set of known device certificates acquired by the CM.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0]

Table 51 - CmCert Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
DeviceId	String	Key		N/A
DeviceCert	DocsX509ASN1DEREncodedCertificate	R/O		N/A

8.5.3.12.1 DeviceId

This key attribute represents the unique identifier of an instance of this object.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0]

8.5.3.12.2 DeviceCert

This attribute represents the X509 DER-encoded CM device certificate.

Reference: Cable Modem Certificate Storage and Management in the CM [SECv4.0], docsBpi2CmDeviceCmCert [RFC 4131] (Legacy PKI CM certificate functions), docsBpi2Ext31CmDeviceCmCert [DOCS-BPI2EXT-MIB] (DOCSIS 3.1/4.0 PKI CM certificate functions)

8.5.3.13 LearnedCmtsCert

The LearnedCmtsCert object describes the set of CMTS device certificates learned by the CM during the BPI+V2 process.

Reference: Baseline Privacy Key Management (BPKM) protocol [SECv4.0].

Table 52 - LearnedCmtsCert Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
DeviceId	String	R/W		N/A
DeviceCert	DocsX509ASN1DEREncodedCertificate	R/W		N/A

8.5.3.13.1 DeviceId

This attribute represents the unique identifier of an instance of this object.

Reference: Baseline Privacy Key Management (BPKM) Protocol [SECv4.0]

8.5.3.13.2 DeviceCert

This attribute represents the X509 DER-encoded CMTS device certificate.

Reference: Baseline Privacy Key Management (BPKM) Protocol [SECv4.0]

8.5.3.14 CmCertRevocationCfg

The CmCertRevocationCfg object describes the revocation control on the CM.

Reference: Authorization Messages Authentication Overview [SECv4.0]

Table 53 - CmCertRevocationCfg Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
CmOnlineCertStatusProtocolCfg	Directed composition to CmOnlineCertStatusProtocolCfg	1	1	

Table 54 - CmCertRevocationCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Method	Enum	R/W	none(1), ocsp(2)	N/A	'none'

8.5.3.14.1 Method

This attribute indicates the used method for performing the revocation checking.

'none' means the revocation checking is disabled on the CM.

'none' - Revocation checking is disabled.

'ocsp' - Use OCSP for revocation checking.

Reference: Authorization Messages Authentication Overview [SECv4.0]

8.5.3.15 CmOnlineCertStatusProtocolCfg

The CmOnlineCertStatusProtocolCfg object provides the configuration information for CM to perform revocation checking using Online Cert Status Protocol.

Reference: Authorization Messages Authentication Overview [SECv4.0]

Table 55 - CmOnlineCertStatusProtocolCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Cache	Enum	R/W	none(1), good(2), revoked(3), unknown(4)	N/A	'none'
RefreshInterval	UnsignedInt	R/W	1..524160	N/A	

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Url	String	R/W	1..2048	N/A	

8.5.3.15.1 Cache

This attribute indicates the cached OCSP status used by CM. If the value of CmCertRevocationCfg::Method is 'none', the CM does not need to check this attribute.

'none' - no OCSP cache

'good' - Certificates in the received CMTS certificate chain are not revoked.

'revoked' - At least one certificate in the received CMTS certificate chain is revoked.

'unknown' - unknown revocation status

Reference: Authorization Messages Authentication Overview [SECv4.0]

8.5.3.15.2 RefreshInterval

This attribute indicates the refresh interval (minutes) for the cached OCSP status used by the CM. If the value of CmCertRevocationCfg::Method is 'none', the CM does not need to check this attribute.

Reference: Authorization Messages Authentication Overview [SECv4.0]

8.5.3.15.3 Url

This attribute is the URL for OCSP communications in checking a certificate's revocation status. This attribute configures the CM with a URL string to retrieve OCSP information. If the value of this attribute is a zero-length string, the CM does not need to check the revocation status of a CMTS device certificate. If the value of CmCertRevocationCfg::Method is 'none', the CM does not need to check this attribute.

Reference: Authorization Messages Authentication Overview [SECv4.0]

8.5.3.16 CertManagementStatus

The CertManagementStatus object provides the information of the certificate status on the CM.

Reference: Authorization Messages Authentication Overview [SECv4.0]

Table 56 - CertManagementStatus Object Attributes

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
DeviceCertStatus	Directed composition to DeviceCertStatus	1	*	
TrustAnchorCertStatus	Directed composition to TrustAnchorCertStatus	1	*	

8.5.3.17 DeviceCertStatus

The DeviceCertStatus object provides the status of the Device Certificates on the CM.

Reference: Authorization Messages Authentication Overview [SECv4.0]

Table 57 - DeviceCertStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Index	UnsignedInt	Key		N/A	
Subject	String	R/O	SIZE (0..255)	N/A	

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Source	Enum	R/O	snmp(1) configFile(2) other(3)	N/A	
DeviceCert	DocsX509ASN1DEREncodedCertificate	R/O		N/A	
SignedCaCert	DocsX509ASN1DEREncodedCertificate	R/O		N/A	

8.5.3.17.1 Index

This key attribute represents the unique identifier of an instance of this object.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.17.2 Subject

This attribute is the subject name encoded in the Device certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.17.3 Source

This attribute indicates the source of the Device certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.17.4 DeviceCert

This attribute represents the X509 DER-encoded CM Device certificate or the learned CMTS Device certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.17.5 SignedCaCert

This attribute represents the X509 DER-encoded CA certificate that signed the CM Device certificate or the learned CMTS Device certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.18 TrustAnchorCertStatus

The TrustAnchorCertStatus object provides the status of the CA certificates on the CM.

Reference: Authorization Messages Authentication Overview [SECv4.0]

Table 58 - TrustAnchorCertStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Index	UnsignedInt	Key		N/A	
Subject	String	R/O	SIZE (0..255)	N/A	
Source	Enum	R/O	snmp(1), configFile(2), bpkm(3), other(4)	N/A	
CaCert	DocsX509ASN1DEREncodedCertificate	R/O		N/A	

8.5.3.18.1 Index

This key attribute represents the unique identifier of an instance of this object.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.18.2 Subject

This attribute is the subject name encoded in the CA certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.18.3 Source

This attribute indicates the source of the CA certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.18.4 CaCert

This attribute represents the X509 DER-encoded CA certificate.

Reference: BPI+ X.509 Certificate Profile and Management [SECv4.0]

8.5.3.19 SsdManagement

The SsdManagement object provides the control of Secure Software Download (SSD) on the CM.

Table 59 - SsdManagement Object Attributes

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
SsdCfg	Directed composition to SsdCfg	1	1	
CodeDownloadControl	Directed composition to CodeDownloadControl	1	1	
SsdStatus	Directed composition to SsdStatus	1	1	

8.5.3.20 SsdCfg

The SsdCfg object contains Secure Software Download (SSD) control attributes for the CM.

Reference: docsDevSoftware [RFC 4639]

Table 60 - SsdCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
SwFilename	String	R/W	SIZE (0..64)	N/A
SwServerAddress	IpAddress	R/W		N/A
TransportProtocol	Enum	R/W	tftp(1), http(2)	N/A
SwAdminControl	Enum	R/W	upgradeFromMgt(1), allowProvisioningUpgrade(2), ignoreProvisioningUpgrade(3)	N/A

8.5.3.20.1 SwFileName

This attribute is the filename of the software image to be downloaded.

Reference: docsDevSwFilename [RFC 4639]

8.5.3.20.2 SwServerAddress

This attribute is the address of the TFTP or HTTP server used for software upgrades. If the TFTP/HTTP server is unknown, return the zero-length address string.

Reference: docsDevSwServerAddress [RFC 4639]

8.5.3.20.3 TransportProtocol

This attribute is the transport protocol (TFTP or HTTP) to be used for software upgrades.

Reference: docsDevSwServerTransportProtocol [RFC 4639]

8.5.3.20.4 SwAdminControl

This attribute is the status of the SSD administration. The possible values for this attribute are listed below:

'upgradeFromMgt' - If set to upgradeFromMgt, the device will initiate a TFTP or HTTP software image download. This indicates that a software download is currently in progress, and that the device will reboot after successfully receiving an image. If the download process is interrupted (e.g., by a reset or power failure), the device will load the previous image and, after re-initialization, continue to attempt loading the image specified in docsDevSwFilename.

'allowProvisioningUpgrade' - If set to allowProvisioningUpgrade, the device will use the software version information supplied by the provisioning server when next rebooting (this does not cause a reboot).

'ignoreProvisioningUpgrade' - If set to ignoreProvisioningUpgrade, the device will disregard software image upgrade information from the provisioning server.

Reference: docsDevSwAdminStatus [RFC 4639]

8.5.3.21 CodeDownloadControl

The CodeDownloadControl object contains code download control attributes for the CM.

Reference: docsBpi2Ext31CodeDownloadControl [DOCS-BPI2EXT-MIB]

Table 61 - CodeDownloadControl Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
UpdateCvcChain	HexBinary	R/W		N/A
MfgOrgName	AdminString	R/O		N/A
MfgCodeAccessStart	DateTime	R/O	SIZE(11)	N/A
MfgCvcAccessStart	DateTime	R/O	SIZE(11)	N/A
CoSignerOrgName	AdminString	R/O		N/A
CoSignerCodeAccessStart	DateTime	R/O	SIZE(11)	N/A
CoSignerCvcAccessStart	DateTime	R/O	SIZE(11)	N/A
OcspCvcResponse	String	R/W	SIZE(0..255)	N/A

8.5.3.21.1 UpdateCvcChain

This attribute is a degenerate PKCS7 signedData structure that contains the CVC and the CVC CA certificate chain in the certificates field. Setting this object triggers the device to verify the CVC and update the cvcAccessStart values associated with the new PKI defined by DOCSIS 4.0. The content of this object is then discarded. If the device is not enabled to upgrade code files, or if the CVC verification fails, the CVC will be rejected. Reading this object always returns the zero-length OCTET STRING.

Reference: docsBpi2Ext31CodeUpdateCvcChain [DOCS-BPI2EXT-MIB]

8.5.3.21.2 MfgOrgName

This attribute is the device manufacturer's organizationName used to validate the code verification certificate issued from the new PKI defined in DOCSIS 4.0.

Reference: docsBpi2Ext31CodeMfgOrgName [DOCS-BPI2EXT-MIB]

8.5.3.21.3 MfgCodeAccessStart

This attribute is the device manufacturer's current codeAccessStart value used with the new PKI defined in DOCSIS 3.1. This value will always refer to Greenwich Mean Time (GMT), and the value is required to contain TimeZone information (fields 8-10).

Reference: docsBpi2Ext31CodeMfgCodeAccessStart [DOCS-BPI2EXT-MIB]

8.5.3.21.4 MfgCvcAccessStart

This attribute is the device manufacturer's current cvcAccessStart value used with the new PKI defined in DOCSIS 4.0. This value will always refer to Greenwich Mean Time (GMT), and the value is required to contain TimeZone information (fields 8-10).

Reference: docsBpi2Ext31CodeMfgCvcAccessStart [DOCS-BPI2EXT-MIB]

8.5.3.21.5 CoSignerOrgName

This attribute is the co-signer's organizationName used to validate the code verification certificate issued from the new PKI defined in DOCSIS 4.0. The value is a zero-length string if the co-signer is not specified.

Reference: docsBpi2Ext31CodeCoSignerOrgName [DOCS-BPI2EXT-MIB]

8.5.3.21.6 CoSignerCodeAccessStart

This attribute is the co-signer's current codeAccessStart value used with the new PKI defined in DOCSIS 4.0. This value will always refer to Greenwich Mean Time (GMT), and the value is required to contain TimeZone information (fields 8-10). If CoSignerOrgName is a zero-length string, the value of this object is meaningless.

Reference: docsBpi2Ext31CodeCoSignerCodeAccessStart [DOCS-BPI2EXT-MIB]

8.5.3.21.7 CoSignerCvcAccessStart

This attribute is the co-signer's current cvcAccessStart value used with the new PKI defined in DOCSIS 4.0. This value will always refer to Greenwich Mean Time (GMT), and the value is required to contain TimeZone information (fields 8-10). If CoSignerOrgName is a zero-length string, the value of this object is meaningless.

Reference: docsBpi2Ext31CodeCoSignerCvcAccessStart [DOCS-BPI2EXT-MIB]

8.5.3.21.8 OcspCvcResponse

This optional attribute is the Revocation Information for the CVC chains.

Reference: Network Initialization [SECv4.0]

8.5.3.22 SsdStatus

The SsdStatus object contains Secure Software Download (SSD) Status attributes for the CM.

Table 62 - SsdStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
SwOperStatus	Enum	R/O	inProgress(1), completeFromProvisioning(2), completeFromMgt(3), failed(4), idle(5), cvcVerified(6), cvcRejected(7), codeFileVerified(8), codeFileRejected(9), unknown(10)	N/A
SwCurrentVers	AdminString	R/O	SIZE(11)	N/A

Attribute Name	Type	Access	Type Constraints	Units
SwStatusCode	Enum	R/O	configFileCvcVerified(1), configFileCvcRejected(2), snmpCvcVerified(3), snmpFileCvcRejected(4), codeFileVerified(5), codeFileRejected(6), other(7)	N/A
SwStatusString	String	R/O		N/A

8.5.3.22.1 SwOperStatus

This attribute is the status of the SSD operation. The possible values for this attribute are listed below:

'inProgress' - This status indicates that a TFTP or HTTP download is underway, either as a result of a version mismatch at provisioning or as a result of a upgradeFromMgt request.

'completeFromProvisioning' - This status indicates that the last software upgrade was a result of version mismatch at provisioning.

'completeFromMgt' - This status indicates that the last software upgrade was a result of setting docsDevSwAdminStatus to upgradeFromMgt.

'failed' - This status indicates that the last attempted download failed, ordinarily due to TFTP or HTTP timeout.

'idle' - This status indicates that the CM is not running a TFTP or HTTP download.

'cvcVerified' - This status indicates that the CVC chain has been verified and is valid.

'cvcRejected' - This status indicates that the CVC chain has been rejected by the CM.

'codeFileVerified' - This status indicates that the code file has been verified and is valid.

'codeFileRejected' - This status indicates that the code file has been rejected by the CM.

'unknown' - unknown status

Reference: docsDevSwOperStatus [RFC 4639]

8.5.3.22.2 SwCurrentVers

This attribute is the software version currently operating in the CM.

Reference: docsDevSwCurrentVers [RFC 4639]

8.5.3.22.3 SwStatusCode

This attribute indicates the result of the latest config file CVC verification, SNMP CVC verification, or code file verification.

'configFileCvcVerified' - This status indicates that the CVC file received via config file has been verified.

'configFileCvcRejected' - This status indicates that the CVC file received via config file has been rejected.

'snmpCvcVerified' - This status indicates that the CVC file received via SNMP has been verified.

'snmpFileCvcRejected' - This status indicates that the CVC file received via SNMP has been rejected.

'codeFileVerified' - This status indicates the code file has been verified.

'codeFileRejected' - This status indicates the code file has been rejected.

'other' - other status

Reference: docsBpi2CodeDownloadStatusCode [RFC 4131]

8.5.3.22.4 SwStatusString

The value of this attribute indicates the additional information to the status code. The value will include the error code and error description, which will be defined separately.

Reference: docsBpi2CodeDownloadStatusString [RFC 4131]

8.5.3.23 SshKeyManagement

The SshKeyManagement object serves as the root for the CM's SSH Key Management functions.

Reference: docsSecCmSshKeyManagement [DOCS-SEC-MIB]

Table 63 - SshKeyManagement Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
SshServer	Directed composition to SshServer	1	1	

8.5.3.24 SshServer

The SshServer object contains information about the CM Secure Shell server function.

Reference: docsSecCmSshServer [DOCS-SEC-MIB]

Table 64 - SshServer Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
CdsFileServer	Directed composition to CdsFileServer	1	1	
SshCmCds	Directed composition to CdsCfg	1	1	
SccaServerCfg	Directed composition to SccaServerCfg	1	1	

Table 65 - SshServer Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
EnabledInterfaces	EnumBits	R/W	customerFacing(0), operatorFacing(1)	N/A	'operatorFacing'
Status	Enum	R/O	disconnectedNotAllowed(1), disconnectedProtocolError(2), disconnectedKeyExchangeFailed(3), disconnectedReserved(4), disconnectedMacError(5), disconnectedCompressionError(6), disconnectedServiceNotAvailable(7), disconnectedProtocolVersionNotSupported(8), disconnectedHostKeyNotVerifiable(9), disconnectedConnectionLost(10), disconnectedByApplication(11), disconnectedTooManyConnections(12), disconnectedAuthCancelledByUser(13), disconnectedNoMoreAuthMethods(14), disconnectedIllegalUserName(15), connected(16), disconnectedUnknown(17)	N/A	
PublicKey	HexBinary	R/O		N/A	

Attribute Name	Type	Access	Type Constraints	Units	Default Value
NewConnectionTimeout	UnsignedShort	R/W	0..28800	seconds	0
InactivityTimeout	UnsignedShort	R/W	0..86400	seconds	1800
SshSourceAddrRestriction	IpAddress	R/W			
SshSourcePrefixRestriction	InetAddressPrefixLength	R/W			

8.5.3.24.1 EnabledInterfaces

This attribute configures which interfaces the SSH server function is enabled on in the CM.

If the 'customerFacing' bit (bit 0) is set to '1', the CM allows access from all local (customer premises) network interfaces/addresses. This includes Ethernet, wireless and MOCA interfaces.

If the 'operatorFacing' bit (bit 1) is set to '1', the CM allows access from all network-facing private interfaces/addresses (i.e., operator's network).

If both Bit 0 and Bit 1 are set, the CM allows access from both local and private network interfaces/addresses.

See section A.1.2.1.3, SSH Enabled Interfaces, in [SECv4.0] for additional details.

Reference: docsSecCmSshServerEnabledInterfaces [DOCS-SEC-MIB]

8.5.3.24.2 Status

This attribute reports the status of the connection between the CM SSH server and the SSH client. The possible values for this attribute are defined in [RFC 4253] and are listed below:

'disconnectedNotAllowed' - SSH client is not allowed to connect to the host.

'disconnectedProtocolError' - SSH client disconnected because of SSH protocol error.

'disconnectedKeyExchangeFailed' - SSH client disconnected because the SSH key exchange failed at the SSH transport layer.

'disconnectedReserved' - Value reserved for future use.

'disconnectedMacError' - SSH client disconnected due to the incompatibility of the Message Authentication code algorithm or value.

'disconnectedCompressionError' - SSH client disconnected due to the failure of compression on the packet payload when it is required or the incompatibility of the compression algorithm exists.

'disconnectedServiceNotAvailable' - SSH client disconnected because SSH service is not available on the server.

'disconnectedProtocolVersionNotSupported' - SSH client disconnected because the SSH protocol version is not supported by the server.

'disconnectedHostKeyNotVerifiable' - SSH client disconnected because of using an unverifiable host key.

'disconnectedConnectionLost' - SSH client disconnected because of inactivity.

'disconnectedByApplication' - SSH server disconnected by the SCCA application when performing the TLS-based Authentication.

'disconnectedTooManyConnections' - SSH client disconnected because the connections limitation has been exceeded.

'disconnectedAuthCancelledByUser' - SSH client disconnected because the authentication is cancelled by the user.

'disconnectedNoMoreAuthMethods' - SSH client disconnected because no more authentication methods are available.

'disconnectedIllegalUserName' - SSH client disconnected because of an illegal username.

'connected' - Connection between the CM SSH server and the SSH client is active.

'disconnectedUnknown' - SSH client disconnected for unknown or other reason.

Reference: docsSecCmSshServerStatus [DOCS-SEC-MIB]

8.5.3.24.3 PublicKey

This attribute is the authorized SSH client public key used by the CM to authenticate the client when the client attempts to set up a CLI SSH connection.

Reference: docsSecCmSshServerPublicKey [DOCS-SEC-MIB]

8.5.3.24.4 NewConnectionTimeout

This attribute is the new SSH connection timeout provisioned on the CM. When this timeout value is reached, the CM sets the Enabled attribute to 'false' and stops accepting new SSH connections. Established connections remain active.

Reference: docsSecCmSshServerNewConnectionTimeout [DOCS-SEC-MIB]

8.5.3.24.5 InactivityTimeout

This attribute is the SSH inactivity timeout provisioned on the CM. This attribute represents the time at which an established connection is terminated if there is no activity. Inactivity is defined as the remote side of the connection timing out and disconnecting. If this attribute is set to zero, the inactivity timeout will be implementation-specific.

Reference: docsSecCmSshServerInactivityTimeout [DOCS-SEC-MIB]

8.5.3.24.6 SshSourceAddrRestriction

This optional attribute is the SSH source address restriction provisioned on the CM. When this attribute is not present, the CM enables unrestricted access to the SSH server.

Reference: docsSecCmSshServerSshSourceAddrRestriction [DOCS-SEC-MIB]

8.5.3.24.7 SshSourcePrefixRestriction

This optional attribute is the SSH source address prefix restriction provisioned on the CM. This attribute is a network/address specifier in CIDR notation that limits the IP addresses where SSH connections can originate.

Reference: docsSecCmSshServerSshSourcePrefixRestriction [DOCS-SEC-MIB]

8.5.3.25 CdsFileServer

The CdsFileServer object contains information the CM uses to access a CDS server in the operator's network.

Reference: docsSecCmCdsFileServer [DOCS-SEC-MIB]

Table 66 - CdsFileServer Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IpAddr	IpAddress	R/W		N/A	
SshCmCdsDownloadUrl	String	R/W	1..2048	N/A	
RevocationStatusAction	Enum	R/W	continue(0), reject(1)	N/A	'continue'

8.5.3.25.1 IpAddr

This attribute is the Internet address of the CDS server in the operator's network.

Reference: docsSecCmCdsFileServerIpAddr [DOCS-SEC-MIB]

8.5.3.25.2 SshCmCdsDownloadUrl

This attribute is the URL of the CDS server in the operator's network.

Reference: docsSecCmCdsFileServerSshCmCdsDownloadUrl [DOCS-SEC-MIB]

8.5.3.25.3 RevocationStatusAction

This attribute is the action taken by the CM if it does not receive revocation status from the provisioning system server. The possible values for this object are listed below:

'continue' - Continue operation with the CDS server

'reject' - Reject the connection with the CDS server

Reference: docsSecCmCdsFileServerRevocationStatusAction [DOCS-SEC-MIB]

8.5.3.26 SshCmCds

The SshCmCds object serves as the root for the CM's SSH Key Management function for configuring the Credential Data Structure (CDS) for the SNMP-based authentication method.

Reference: docsSecCmSshCmCds [DOCS-SEC-MIB]

Table 67 - SshCmCds Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
PasswordCredential	Directed composition to PasswordCredential	1	0..*	
PublicKeyCredential	Directed composition to PublicKeyCredential	1	0..*	

8.5.3.27 PasswordCredential

The PasswordCredential object contains authorized password credential information for the CM to authenticate SSH Client CLI connections.

The CM MUST support creation of new instances of the PasswordCredential object and deletion of existing PasswordCredential object instances.

Reference: docsSecCmPasswordCredentialTable [DOCS-SEC-MIB]

Table 68 - PasswordCredential Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
Index	UnsignedInt	Key		N/A
UserId	AdminString	R/W		N/A
Password	HexBinary	R/W	8..128	N/A
MacAddr	MacAddress	R/W		N/A

8.5.3.27.1 Index

This key attribute represents the unique identifier of an instance of this object.

Reference: docsSecCmPasswordCredentialIndex [DOCS-SEC-MIB]

8.5.3.27.2 UserId

This attribute is the identifier of the user for which the password credential is to be evaluated.

Reference: docsSecCmPasswordCredentialUserId [DOCS-SEC-MIB]

8.5.3.27.3 Password

This attribute is a string encoded in the [ISO8859-1] character-set and using characters in the range from 0x21 – 0x7E and serving as the credential to be evaluated for the user.

Reference: docsSecCmPasswordCredentialPassword [DOCS-SEC-MIB]

8.5.3.27.4 MacAddr

This optional attribute is the MAC address assigned to the CM.

Reference: docsSecCmPasswordCredentialMacAddr [DOCS-SEC-MIB]

8.5.3.28 PublicKeyCredential

The PublicKeyCredential object contains authorized RSA Public Key credential information for the CM to authenticate SSH Client CLI connections.

The CM MUST support creation of new instances of the PublicKeyCredential object and deletion of existing PublicKeyCredential object instances.

Reference: docsSecCmPublicKeyCredentialTable [DOCS-SEC-MIB]

Table 69 - PublicKeyCredential Object Attributes

Attribute Name	Type	Access	Type Constraints	Units
Index	UnsignedInt	Key		N/A
SshPublicKey	HexBinary	R/W	256..512	N/A
MacAddr	MacAddress	R/W		N/A

8.5.3.28.1 Index

This key attribute represents the unique identifier of an instance of this object.

Reference: docsSecCmPublicKeyCredentialIndex [DOCS-SEC-MIB]

8.5.3.28.2 SshPublicKey

This attribute is a string containing a DER-encoded RSA or ECDSA public keys in ASN.1 type, as defined in [X.509] and serving as the credential to be evaluated for the user.

Reference: docsSecCmPublicKeyCredentialSshPublicKey [DOCS-SEC-MIB]

8.5.3.28.3 MacAddr

This optional attribute is the MAC address assigned to the CM.

Reference: docsSecCmPublicKeyCredentialMacAddr [DOCS-SEC-MIB]

8.5.3.29 SccaServerCfg

The SccaServerCfg object contains the information the CM uses to access to SCCA API on HTTPS server in the operator's network.

Reference: docsSecCmSccaServerCfg [DOCS-SEC-MIB]

Table 70 - SccaServerCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IpAddr	IpAddress	R/W		N/A	
RestApiUrl	String	R/W	1..2048	N/A	
RevocationStatusAction	Enum	R/W	continue(0), reject(1),	N/A	'continue'

8.5.3.29.1 IpAddr

This attribute is the Internet address of the CDS server in the operator's network.

Reference: docsSecCmSccaServerCfgIpAddr [DOCS-SEC-MIB]

8.5.3.29.2 RestApiUrl

This attribute is the URL of the SCCA REST API to validate the user credentials.

Reference: docsSecCmSccaServerCfgRestApiUrl [DOCS-SEC-MIB]

8.5.3.29.3 RevocationStatusAction

This attribute configures the action taken by the CM if it does not receive revocation status from the HTTPS server. The possible values for this object are listed below:

'continue' - Continue operation with the HTTPS server

'reject' - Reject the connection with the HTTPS server

Reference: docsSecCmSccaServerCfgRevocationStatusAction [DOCS-SEC-MIB]

9 OSSI FOR CMCI

This section defines the operational mechanisms needed to support the transmission of data over cable services between a Cable Modem (CM) and Customer Premise Equipment (CPE). Specifically, this section outlines the following:

- SNMP access via CMCI
- Console Access
- CM diagnostic capabilities
- Protocol Filtering
- Required MIBs

Refer to Section 6 of [CMCIv3.0] for additional CMCI requirements.

9.1 SNMP Access via CMCI

DOCSIS 4.0 CMs have provisions for dual-stack management or management of the CM using SNMP over IPv4 and IPv6. During provisioning, the management of the CM is determined by the MSO. However, SNMP access from the CMCI port(s) for diagnostic purposes prior to the CM being registered needs to operate in a dual-stack management mode and allow access for both IPv4 and IPv6 hosts. CM SNMP access from the CMCI before completing the CMTS registration process MUST comply with the access requirements specified in Section 8.5.2.1. The CM DHCP-acquired IP MUST ignore SNMP requests from CMCI before registration.

The CM DHCP-acquired IP MUST accept SNMP requests from CMCI after completing the CMTS registration process where such SNMP access complies with the requirements stated in Section 8.5.2.2.

The CM MUST support SNMP access, as specified in Section 8.5.2, through the following IP addresses regardless of the CM registration state:

- The CM MUST support 192.168.100.1, as the well-known diagnostic IP address accessible only from the CMCI interfaces. The CM MUST support the well-known diagnostic IP address, 192.168.100.1, on all physical interfaces associated with the CMCI. The CM MUST drop SNMP requests coming from the RF interface targeting the well-known IP address.
- The CM MAY also implement alternative IPv4 interfaces like link-local method described in [RFC 3927]. If implemented, the CM MUST restrict the IP address range described in "Address Selection, Defense and Delivery" of [RFC 3927] to 169.254.1.0 to 169.254.254.255 inclusive.
- The CM MAY support an IPv6 EUI-64 link-local scope address in the format FE80::<*vendorId*>:FFFE:<*remainingMacAddress*> of the CMCI port. The CM MUST drop SNMP requests coming from the RF interface targeting this IPv6 address. Refer to [RFC 4291] for additional details.

9.2 Console Access

The CM MUST NOT allow access to the CM functions by a console port. In this specification, a console port is defined as a communication path, either hardware or software, that allows a user to issue commands to modify the configuration or operational status of the CM. The CM MUST only allow access using DOCSIS defined RF interfaces and operator-controlled SNMP access by the CMCI.

9.3 CM Diagnostic Capabilities

The CM MAY have a diagnostic interface for debugging and troubleshooting purposes. If supported, the CM's diagnostic interface MUST be limited by default to the requirements described in Section 8.5.2 before and after registration. The CM's diagnostic interface SHOULD be disabled by default after registration has been completed.

The CM MAY provide additional controls that will enable the MSO to alter or customize the diagnostic interface, such as by the configuration process or management through the setting of a proprietary MIB.

9.4 Protocol Filtering

The CM MUST be capable of filtering traffic to and from the host CPE as defined in Annex B.

10 OSSI FOR LED INDICATORS

The CM SHOULD support standard front-panel LEDs (Light Emitting Diodes) that present straightforward information about the registration state of the CM so as to facilitate efficient customer support operations.

10.1 CM LED Requirements and Operation

A CM SHOULD support LEDs which have three states: 1) unlit, 2) flash, 3) lit solid. A CM LED in the 'flash' state SHOULD turn on and off with a 50% duty cycle at a frequency not less than 2 cycles per second. A CM SHOULD support LEDs which light sequentially, following the normal CM initialization procedure specified in [MULPIv4.0]. In this way, the installer can detect a failure that prevents the CM from becoming operational.

A CM SHOULD have a minimum of five externally visible LEDs divided into three functional groups as indicated below:

BOX: This group SHOULD have 1 LED labeled as POWER for the BOX status.

DOCSIS: This group SHOULD have 3 LEDs labeled as DS, US, and ONLINE for the DOCSIS interface status. The LEDs in the DOCSIS group SHOULD be in the order: DS, US, and ONLINE, from left to right, or top to bottom, as appropriate for the orientation of the device.

CPE: This group SHOULD have a minimum of 1 LED labeled as LINK for the LINK status. The CM MAY have multiple LEDs in the CPE group to represent individual CPE interface types and parameters. These CM CPE LEDs MAY be labeled according to their associated interface types.

There is no specific requirement for labeling the functional groups. The overall CM LED distribution SHOULD be in the order: POWER, DS, US, ONLINE, and LINK.

The CM SHOULD use these LEDs to indicate that the following modes of operation are in progress, or have completed successfully:

- Power on, Software Application Image Validation and Self Test
- Scan for Downstream Channel
- Resolve CM-SG and Range
- Operational
- Data Link and Activity

The CM SHOULD operate its LEDs as described in the following sections for each of the above modes of operation.

10.1.1 Power On, Software Application Image Validation and Self Test

The CM SHOULD, when turned on, place the LEDs, or at least the DOCSIS Group LEDs (DS, US, ONLINE), in the 'flash' state while the CM performs the system initialization of the Operational System, CM application load, and any proprietary self-tests. Following the successful completion of the steps above, the CM SHOULD place the LEDs, or at least the DOCSIS Group LEDs, in the 'lit solid' state for one second, after which the CM places the POWER LED in the 'lit solid' state. The CM MAY also place the LINK LED in the 'lit solid' state if a CPE device is properly connected (see Section 10.1.5). If the system initialization, described above, results in a failure, the CM SHOULD place the LEDs, or at least the DOCSIS Group LEDs in the 'flash' state, in which they should remain.

10.1.2 Scan for Downstream Channel

The CM SHOULD place the DS LED in the 'flash' state as the CM scans for a candidate primary downstream DOCSIS channel. The CM SHOULD place the DS LED in the 'lit solid' state when the CM MAC layer has completed synchronization of MPEG framing of the candidate primary downstream channel, as defined in the "Cable Modem Initialization and Reinitialization" section of [MULPIv4.0]. The CM SHOULD maintain the 'lit solid' state of the DS LED as the CM continues the initialization process. The CM SHOULD NOT place the DS

LED in the 'flash' state when resolving the CM service groups or performing downstream acquisition of CM receive channels in the registration process as defined in the "Cable Modem Initialization and Reinitialization" section of [MULPIv4.0].

Whenever the CM restarts CM initialization (which can include scanning for a downstream channel and attempting to synchronize to a downstream channel), the CM SHOULD place the DS LED in the 'flash' state and the US LED and ONLINE LED in the 'unlit' state.

10.1.3 Resolve CM-SG and Range

After the CM places the DS LED in the 'lit solid' state, the CM SHOULD place the US LED in the 'flash' state and the ONLINE LED in the 'unlit' state while the CM is determining CM-SGs and performing initial ranging, until the CM receives a ranging response message with a ranging status of 'success' from the CMTS. When the CM receives a ranging response message with a ranging status of 'success' from the CMTS, the CM SHOULD place the US LED in the 'lit solid' state.

The CM SHOULD maintain the 'lit solid' state of the US LED as the CM continues the initialization process. Unless the channel used to transmit the registration request message is not in the TCC received in the registration response message, the CM SHOULD NOT place the US LED in the 'flash' state when performing upstream acquisition of CM transmit channels in the registration process as defined in the "Cable Modem Initialization and Reinitialization" section of [MULPIv4.0]. The CM SHOULD maintain the 'lit solid' state of the US LED when the CM is ranged on one or more upstream channels.

10.1.4 Operational

After the CM places the US LED in the 'lit solid' state, the CM SHOULD place the ONLINE LED in the 'flash' state while the CM continues the process towards become operational (this includes performing early authentication, establishing IP connectivity, and registering with the CMTS, and performing BPI initialization). When the CM is operational, the CM SHOULD place the ONLINE LED in the 'lit solid' state. Operational is defined according to section "Cable Modem Initialization and Reinitialization" in [MULPIv4.0].

If at any point there is a failure in the registration process that causes the CM to lose its operational state, including but not limited to loss of the primary downstream channel, ranging, DHCP, configuration file download, registration, and Baseline Privacy initialization, the CM SHOULD place the ONLINE LED in the 'flash' state.

If the CM becomes operational and the CM configuration file has the Network Access Control Object (NACO) set to zero (0), the CM SHOULD place the ONLINE LED in the 'unlit' state and place both the 'DS and US LEDs in the 'flash' state. Refer to the Common Radio Frequency Interface Encodings Annex of [MULPIv4.0] for details on the Network Access Control Object (NACO).

10.1.5 Data Link and Activity

The CM SHOULD place the LINK LED in the 'lit solid' state when a CPE device is connected and the CM is not bridging data. The CM SHOULD place the LINK LED in the 'flash' state ONLY when the CM is bridging data during the CM operational state and NACO set to one (1). The CM SHOULD NOT place the LINK LED in the 'flash' state for data traffic originating or terminating at the CM device itself.

If LINK is detected with a CPE device, the CM MAY set the LINK LED to the 'lit solid' state any time after the power and self test steps are completed.

10.2 Additional CM Operational Status Visualization Features

The CM MAY change the DOCSIS defined LED behavior when the CM is in a vendor proprietary mode of operation. The CM MUST NOT have additional LEDs that reveal DOCSIS specific information about the configuration file content, or otherwise clearly specified (see NACO visualization in Sections 10.1.4 and 10.1.5.

10.2.1 Secure Software Download

The CM SHOULD signal that a Secure Software Download is in process, by setting the DS LED and the US LED to the 'flash' state, and the ONLINE LED to the 'lit solid' state.

Annex A Detailed MIB Requirements (Normative)

This Annex defines the SNMP MIB modules and MIB variables required for DOCSIS 4.0 CM devices.

Table 71 - MIB Implementation Support

Requirement Type	Table Notation	Description
Deprecated	D	Deprecated objects are optional. If a vendor chooses to implement the object, the object is required to be implemented correctly according to the MIB definition. If a vendor chooses not to implement the object, an agent is not to instantiate such object and is required to respond with the appropriate error/exception condition (e.g., 'noSuchObject' for SNMPv2c).
Mandatory	M	The object is required to be implemented correctly according to the MIB definition.
Not Applicable	NA	Not applicable to the device.
Not Supported	N-Sup	An agent is not to instantiate such object and is required to respond with the appropriate error/exception condition (e.g., 'noSuchObject' for SNMPv2c).
Optional	O	A vendor can choose to implement or not implement the object. If a vendor chooses to implement the object, the object is required to be implemented correctly according to the MIB definition. If a vendor chooses not to implement the object, an agent is not to instantiate such object and is required to respond with the appropriate error/exception condition (e.g., 'noSuchObject' for SNMPv2c).
Obsolete	Ob	In SNMP convention, obsolete objects are not to be implemented. This specification allows vendors to implement or not implement obsolete objects. If a vendor chooses to implement an obsoleted object, the object is required be implemented correctly according to the MIB definition. If a vendor chooses not to implement the obsoleted object, the SNMP agent is not to instantiate such object and is required to respond with the appropriate error/exception condition (e.g., 'noSuchObject' for SNMPv2c).

Table 72 - SNMP Access Requirements

SNMP Access Type	Table Notation	Description
N-Acc	Not Accessible	The object is not accessible and is usually an index in a table
Read Create	RC	The access of the object is implemented as Read-Create
Read Write	RW	The access of the object is implemented as Read-Write
Read Only	RO	The access of the object is implemented as Read-Only
Read Create or Read Only	RC/RO	The access of the object is implemented as either Read-Create or Read-Only as described in the MIB definition
Read Write / Read Only	RW/RO	The access of the object is implemented as either Read-Write or Read-Only as described in the MIB definition
Accessible for SNMP Notifications	Acc-FN	These objects are used for SNMP Notifications by the CM SNMP Agent

A.1 MIB-Object Details

The CM instantiates SNMP MIB objects based on its configuration and operational parameters acquired during registration. Below are denominations for several Table 73 columns that indicate modes of operation where a CM has specific management requirements for certain MIB object instantiation and syntax.

The CM always operates in "1.1 QoS Mode".

The CM SNMP access control configuration is either NmAccess Mode or SNMP Coexistence Mode.

The CM upstream channel types can be categorized as "OFDMA upstream", "TDMA/ATDMA upstream" and "SCDMA upstream".

Table 73 - MIB Object Details

DOCS-IF-MIB [RFC 4546] **Note:** Refer to Section 7.1.3.7 for detailed requirements for handling SC-QAM and OFDM/OFDMA channels.		
Object	**CM**	**Access**
docsIfDownstreamChannelTable	M	N-Acc
docsIfDownstreamChannelEntry	M	N-Acc
docsIfDownChannelId	M	RO
docsIfDownChannelFrequency	M	RO
docsIfDownChannelWidth	M	RO
docsIfDownChannelModulation	M	RO
docsIfDownChannelInterleave	M	RO
docsIfDownChannelPower	M	RO
docsIfDownChannelAnnex	M	RO
docsIfDownChannelStorageType	M	RO

Object	**CM TDMA/ ATDMA upstream**	**Access**		**CM SCDMA upstrea m**	**Access**
docsIfUpstreamChannelTable	M	N-Acc	**docsIfUpstreamChannelTable**	O	N-Acc
docsIfUpstreamChannelEntry	M	N-Acc	**docsIfUpstreamChannelEntry**	O	N-Acc
docsIfUpChannelId	M	RO	docsIfUpChannelId	O	RO
docsIfUpChannelFrequency	M	RO	docsIfUpChannelFrequency	O	RO
docsIfUpChannelWidth	M	RO	docsIfUpChannelWidth	O	RO
docsIfUpChannelModulationProfile	M	RO	docsIfUpChannelModulationProfile	O	RO
docsIfUpChannelSlotSize	M	RO	docsIfUpChannelSlotSize	O	RO
docsIfUpChannelTxTimingOffset	M	RO	docsIfUpChannelTxTimingOffset	O	RO
docsIfUpChannelRangingBackoffStart	M	RO	docsIfUpChannelRangingBackoffStart	O	RO
docsIfUpChannelRangingBackoffEnd	M	RO	docsIfUpChannelRangingBackoffEnd	O	RO
docsIfUpChannelTxBackoffStart	M	RO	docsIfUpChannelTxBackoffStart	O	RO
docsIfUpChannelTxBackoffEnd	M	RO	docsIfUpChannelTxBackoffEnd	O	RO
docsIfUpChannelScdmaActiveCodes	O	RO	docsIfUpChannelScdmaActiveCodes	O	RO
docsIfUpChannelScdmaCodesPerSlot	O	RO	docsIfUpChannelScdmaCodesPerSlot	O	RO
docsIfUpChannelScdmaFrameSize	O	RO	docsIfUpChannelScdmaFrameSize	O	RO
docsIfUpChannelScdmaHoppingSeed	O	RO	docsIfUpChannelScdmaHoppingSeed	O	RO
docsIfUpChannelType	M	RO	docsIfUpChannelType	O	RO
docsIfUpChannelCloneFrom	O	RO	docsIfUpChannelCloneFrom	O	RO
docsIfUpChannelUpdate	O	RO	docsIfUpChannelUpdate	O	RO
docsIfUpChannelStatus	O	RO	docsIfUpChannelStatus	O	RO
docsIfUpChannelPreEqEnable	M	RO	docsIfUpChannelPreEqEnable	O	RO

Object	**CM**	**Access**
docsIfQosProfileTable	D	N-Acc
docsIfQosProfileEntry	D	N-Acc
docsIfQosProfIndex	D	N-Acc
docsIfQosProfPriority	D	RO

docsIfQosProfMaxUpBandwidth	D	RO
docsIfQosProfGuarUpBandwidth	D	RO
docsIfQosProfMaxDownBandwidth	D	RO
docsIfQosProfMaxTxBurst	D	RO
docsIfQosProfBaselinePrivacy	D	RO
docsIfQosProfStatus	D	RO
docsIfQosProfMaxTransmitBurst	D	RO
docsIfQosProfStorageType	D	RO
Object	**CM**	**Access**
docsIfSignalQualityTable	M	N-Acc
docsIfSignalQualityEntry	M	N-Acc
docsIfSigQIncludesContention	M	RO
docsIfSigQUnerroreds	M	RO
docsIfSigQCorrecteds	M	RO
docsIfSigQUncorrectables	M	RO
docsIfSigQSignalNoise	D	RO
docsIfSigQMicroreflections	M	RO
docsIfSigQEqualizationData	M	RO
docsIfSigQExtUnerroreds	M	RO
docsIfSigQExtCorrecteds	M	RO
docsIfSigQExtUncorrectables	M	RO
Object	**CM**	**Access**
docsIfDocsisBaseCapability	D	RO
docsIfCmMacTable	M	N-Acc
docsIfCmMacEntry	M	N-Acc
docsIfCmCmtsAddress	M	RO
docsIfCmCapabilities	M	RO
docsIfCmRangingRespTimeout	Ob	RW
docsIfCmRangingTimeout	M	RW
docsIfCmStatusTable	D	N-Acc
docsIfCmStatusEntry	D	N-Acc
docsIfCmStatusValue	D	RO
docsIfCmStatusCode	D	RO
docsIfCmStatusTxPower	D	RO
docsIfCmStatusResets	D	RO
docsIfCmStatusLostSyncs	D	RO
docsIfCmStatusInvalidMaps	D	RO
docsIfCmStatusInvalidUcds	D	RO
docsIfCmStatusInvalidRangingResponses	D	RO
docsIfCmStatusInvalidRegistrationResponses	D	RO
docsIfCmStatusT1Timeouts	D	RO
docsIfCmStatusT2Timeouts	D	RO
docsIfCmStatusT3Timeouts	D	RO
docsIfCmStatusT4Timeouts	D	RO
docsIfCmStatusRangingAborteds	D	RO
docsIfCmStatusDocsisOperMode	D	RO
docsIfCmStatusModulationType	D	RO

docsIfCmStatusEqualizationData	D	RO
docsIfCmStatusUCCs	D	RO
docsIfCmStatusUCCFails	D	RO
IF-MIB [RFC 2863]		
Object	**CM**	**Access**
ifNumber	M	RO
ifTableLastChange	M	RO
ifTable **Note:** The ifTable Counter32 objects are not reflected here; refer to Table 76 and Table 77 of A.2 for details on these objects.	M	N-Acc
ifEntry	M	N-Acc
ifIndex	M	RO
ifDescr	M	RO
ifType	M	RO
ifMtu	M	RO
ifSpeed	M	RO
ifPhysAddress	M	RO
ifAdminStatus	M	RW
ifOperStatus	M	RO
ifLastChange	M	RO
ifOutQLen	D	RO
ifSpecific	D	RO
ifXTable **Note:** The ifXTable Counter32 and Counter64 objects are not reflected here; refer to Table 76 and Table 77 of A.2 for details on these objects.	M	N-Acc
ifXEntry	M	N-Acc
ifName	M	RO
ifLinkUpDownTrapEnable	M	RW
ifHighSpeed	M	RO
ifPromiscuousMode	M	RW/RO
ifConnectorPresent	M	RO
ifAlias	M	RW/RO
ifCounterDiscontinuityTime	M	RO
ifStackTable	M	N-Acc
ifStackEntry	M	N-Acc
ifStackHigherLayer	M	N-Acc
ifStackLowerLayer	M	N-Acc
ifStackStatus	M	RC/RO
ifStackLastChange	M	RO
Object	**CM**	**Access**
ifRcvAddressTable	O	N-Acc
ifRcvAddressEntry	O	N-Acc
ifRcvAddressAddress	O	N-Acc
ifRcvAddressStatus	O	RC
IfRcvAddressType	O	RC
Notification		
linkUp	M	Acc-FN

linkDown **Note:** See Section 7.1.3.8.4 for details.	M	Acc-FN
ifTestTable	D	N-Acc
ifTestEntry	D	N-Acc
ifTestId	D	RW
ifTestStatus	D	RW
ifTestType	D	RW
ifTestResult	D	RO
ifTestCode	D	RO
ifTestOwner	D	RW
BRIDGE-MIB [RFC 4188]		
Object	**CM**	**Access**
dot1dBase		
dot1dBaseBridgeAddress	M	RO
dot1dBaseNumPorts	M	RO
dot1dBaseType	M	RO
dot1dBasePortTable	M	N-Acc
dot1dBasePortEntry	M	N-Acc
dot1dBasePort	M	RO
dot1dBasePortIfIndex	M	RO
dot1dBasePortCircuit	M	RO
dot1dBasePortDelayExceededDiscards	M	RO
dot1dBasePortMtuExceededDiscards	M	RO
dot1dStp		
dot1dStpProtocolSpecification	M	RO
dot1dStpPriority	M	RW
dot1dStpTimeSinceTopologyChange	M	RO
dot1dStpTopChanges	M	RO
dot1dStpDesignatedRoot	M	RO
dot1dStpRootCost	M	RO
dot1dStpRootPort	M	RO
dot1dStpMaxAge	M	RO
dot1dStpHelloTime	M	RO
dot1dStpHoldTime	M	RO
dot1dStpForwardDelay	M	RO
dot1dStpBridgeMaxAge	M	RW
dot1dStpBridgeHelloTime	M	RW
dot1dStpBridgeForwardDelay	M	RW
dot1dStpPortTable **Note:** This table is required ONLY if STP is implemented.	O	N-Acc
dot1dStpPortEntry	O	N-Acc
dot1dStpPort	O	RO
dot1dStpPortPriority	O	RW
dot1dStpPortState	O	RO
dot1dStpPortEnable	O	RW
dot1dStpPortPathCost	O	RW
dot1dStpPortDesignatedRoot	O	RO

dot1dStpPortDesignatedCost	O	RO
dot1dStpPortDesignatedBridge	O	RO
dot1dStpPortDesignatedPort	O	RO
dot1dStpPortForwardTransitions	O	RO
dot1dStpPortPathCost32	O	RO
dot1dTp **Note:** This group is required ONLY if transparent bridging is implemented.		
dot1dTpLearnedEntryDiscards	M	RO
dot1dTpAgingTime	M	RW
dot1dTpFdbTable	M	N-Acc
dot1dTpFdbEntry	M	N-Acc
dot1dTpFdbAddress	M	RO
dot1dTpFdbPort	M	RO
dot1dTpFdbStatus	M	RO
dot1dTpPortTable	M	N-Acc
dot1dTpPortEntry	M	N-Acc
dot1dTpPort	M	RO
dot1dTpPortMaxInfo	M	RO
dot1dTpPortInFrames	M	RO
dot1dTpPortOutFrames	M	RO
dot1dTpPortInDiscards	M	RO
dot1dStaticTable **Note:** Implementation of dot1dStaticTable is OPTIONAL.	O	N-Acc
dot1dStaticEntry	O	N-Acc
dot1dStaticAddress	O	RW
dot1dStaticReceivePort	O	RW
dot1dStaticAllowedToGoTo	O	RW
dot1dStaticStatus	O	RW
Notification		
newRoot	O	Acc-FN
topologyChange	O	Acc-FN
DOCS-CABLE-DEVICE-MIB [RFC 4639]		
Object	**CM**	**Access**
docsDevBase		
docsDevRole	M	RO
docsDevDateTime	M	RO/RW
docsDevResetNow	M	RW
docsDevSerialNumber	M	RO
docsDevSTPControl	M	RW/RO
docsDevIgmpModeControl	N-Sup	
docsDevMaxCpe	M	RO

Object	CM in NmAccess Mode	Access	Object	CM in SNMP Coexistence Mode	Access
docsDevNmAccessTable	M	N-Acc	**docsDevNmAccessTable**	N-Sup	
docsDevNmAccessEntry	M	N-Acc	**docsDevNmAccessEntry**	N-Sup	
docsDevNmAccessIndex	M	N-Acc	docsDevNmAccessIndex	N-Sup	

docsDevNmAccessIp	M	RC	docsDevNmAccessIp	N-Sup	
docsDevNmAccessIpMask	M	RC	docsDevNmAccessIpMask	N-Sup	
docsDevNmAccessCommunity	M	RC	docsDevNmAccessCommunity	N-Sup	
docsDevNmAccessControl	M	RC	docsDevNmAccessControl	N-Sup	
docsDevNmAccessInterfaces	M	RC	docsDevNmAccessInterfaces	N-Sup	
docsDevNmAccessStatus	M	RC	docsDevNmAccessStatus	N-Sup	
docsDevNmAccessTrapVersion	M	RC	docsDevNmAccessTrapVersion	N-Sup	

Object	CM	Access
docsDevSoftware		
docsDevSwServer	D	RW
docsDevSwFilename	M	RW
docsDevSwAdminStatus	M	RW
docsDevSwOperStatus	M	RO
docsDevSwCurrentVers	M	RO
docsDevSwServerAddressType	M	RW
docsDevSwServerAddress	M	RW
docsDevSwServerTransportProtocol	M	RW
docsDevServer		
docsDevServerBootState	D	RO
docsDevServerDhcp	D	RO
docsDevServerTime	D	RO
docsDevServerTftp	D	RO
docsDevServerConfigFile	M	RO
docsDevServerDhcpAddressType	M	RO
docsDevServerDhcpAddress	M	RO
docsDevServerTimeAddressType	M	RO
docsDevServerTimeAddress	M	RO
docsDevServerConfigTftpAddressType	M	RO
docsDevServerConfigTftpAddress	M	RO
docsDevEvent		
docsDevEvControl	M	RW
docsDevEvSyslog	D	RW
docsDevEvThrottleAdminStatus	M	RW
docsDevEvThrottleInhibited	D	RO
docsDevEvThrottleThreshold	M	RW
docsDevEvThrottleInterval	M	RW
docsDevEvControlTable	M	N-Acc
docsDevEvControlEntry	M	N-Acc
docsDevEvPriority	M	N-Acc
docsDevEvReporting	M	RW
docsDevEventTable	M	N-Acc
docsDevEventEntry	M	N-Acc
docsDevEvIndex	M	N-Acc
docsDevEvFirstTime	M	RO
docsDevEvLastTime	M	RO

docsDevEvCounts	M	RO
docsDevEvLevel	M	RO
docsDevEvId	M	RO
docsDevEvText	M	RO
docsDevEvSyslogAddressType	M	RW
docsDevEvSyslogAddress	M	RW
docsDevEvThrottleThresholdExceeded	M	RO
docsDevFilter		
docsDevFilterLLCUnmatchedAction	D	RW
docsDevFilterLLCTable	D	N-Acc
docsDevFilterLLCEntry	D	N-Acc
docsDevFilterLLCIndex	D	N-Acc
docsDevFilterLLCStatus	D	RC
docsDevFilterLLCIfIndex	D	RC
docsDevFilterLLCProtocolType	D	RC
docsDevFilterLLCProtocol	D	RC
docsDevFilterLLCMatches	D	RO
Object	**CM**	**Access**
docsDevFilterIpDefault	D	RW
docsDevFilterIpTable	D	N-Acc
docsDevFilterIpEntry	D	N-Acc
docsDevFilterIpIndex	D	N-Acc
docsDevFilterIpStatus	D	RC
docsDevFilterIpControl	D	RC
docsDevFilterIpIfIndex	D	RC
docsDevFilterIpDirection	D	RC
docsDevFilterIpBroadcast	D	RC
docsDevFilterIpSaddr	D	RC
docsDevFilterIpSmask	D	RC
docsDevFilterIpDaddr	D	RC
docsDevFilterIpDmask	D	RC
docsDevFilterIpProtocol	D	RC
docsDevFilterIpSourcePortLow	D	RC
docsDevFilterIpSourcePortHigh	D	RC
docsDevFilterIpDestPortLow	D	RC
docsDevFilterIpDestPortHigh	D	RC
docsDevFilterIpMatches	D	RO
docsDevFilterIpTos	D	RC
docsDevFilterIpTosMask	D	RC
docsDevFilterIpContinue	D	RC
docsDevFilterIpPolicyId	D	RC
docsDevFilterPolicyTable	D	N-Acc
docsDevFilterPolicyEntry	D	N-Acc
docsDevFilterPolicyIndex	D	N-Acc
docsDevFilterPolicyId	D	RC
docsDevFilterPolicyStatus	D	RC
docsDevFilterPolicyPtr	D	RC

docsDevFilterTosTable	D	N-Acc
docsDevFilterTosEntry	D	N-Acc
docsDevFilterTosIndex	D	N-Acc
docsDevFilterTosStatus	D	RC
docsDevFilterTosAndMask	D	RC
docsDevFilterTosOrMask	D	RC
docsDevCpeEnroll	O	RW
docsDevCpeIpMax	O	RW
docsDevCpeTable	Ob	N-Acc
docsDevCpeEntry	Ob	N-Acc
docsDevCpeIp	Ob	N-Acc
docsDevCpeSource	Ob	RO
docsDevCpeStatus	Ob	RC
docsDevCpeInetTable	O	N-Acc
docsDevCpeInetEntry	O	N-Acc
docsDevCpeInetType	O	N-Acc
docsDevCpeInetAddr	O	RC
docsDevCpeInetSource	O	RO
docsDevCpeInetRowStatus	O	RC
IP-MIB [RFC 4293]		
Object	**CM**	**Access**
ipv4GeneralGroup		
ipForwarding	M	RW
ipDefaultTTL	M	RW
ipReasmTimeout	M	RW
ipv6GeneralGroup2		
ipv6IpForwarding	M	RW
ipv6IpDefaultHopLimit	M	RW
ipv4InterfaceTableLastChange	M	RO
ipv4InterfaceTable	M	N-Acc
ipv4InterfaceEntry	M	N-Acc
ipv4InterfaceIfIndex	M	N-Acc
ipv4InterfaceReasmMaxSize	M	RO
ipv4InterfaceEnableStatus	M	RW
ipv4InterfaceRetransmitTime	M	RO
Object	**CM**	**Access**
ipv6InterfaceTableLastChange	M	RO
ipv6InterfaceTable	M	N-Acc
ipv6InterfaceEntry	M	N-Acc
ipv6InterfaceIfIndex	M	N-Acc
ipv6InterfaceReasmMaxSize	M	RO
ipv6InterfaceIdentifier	M	RO
ipv6InterfaceEnableStatus	M	RW
ipv6InterfaceReachableTime	M	RO
ipv6InterfaceRetransmitTime	M	RO
ipv6InterfaceForwarding	M	RW
ipSystemStatsTable	O	N-Acc

ipSystemStatsEntry	O	N-Acc
ipSystemStatsIPVersion	O	N-Acc
ipSystemStatsInReceives	O	RO
ipSystemStatsHCInReceives	O	RO
ipSystemStatsInOctets	O	RO
ipSystemStatsHCInOctets	O	RO
ipSystemStatsInHdrErrors	O	RO
ipSystemStatsInNoRoutes	O	RO
ipSystemStatsInAddrErrors	O	RO
ipSystemStatsInUnknownProtos	O	RO
ipSystemStatsInTruncatedPkts	O	RO
ipSystemStatsInForwDatagrams	O	RO
ipSystemStatsHCInForwDatagrams	O	RO
ipSystemStatsReasmReqds	O	RO
ipSystemStatsReasmOKs	O	RO
ipSystemStatsReasmFails	O	RO
ipSystemStatsInDiscards	O	RO
ipSystemStatsInDelivers	O	RO
ipSystemStatsHCInDelivers	O	RO
ipSystemStatsOutRequests	O	RO
ipSystemStatsHCOutRequests	O	RO
ipSystemStatsOutNoRoutes	O	RO
ipSystemStatsOutForwDatagrams	O	RO
ipSystemStatsHCOutForwDatagrams	O	RO
ipSystemStatsOutDiscards	O	RO
ipSystemStatsOutFragReqds	O	RO
ipSystemStatsOutFragOKs	O	RO
ipSystemStatsOutFragFails	O	RO
ipSystemStatsOutFragCreates	O	RO
ipSystemStatsOutTransmits	O	RO
ipSystemStatsHCOutTransmits	O	RO
ipSystemStatsOutOctets	O	RO
ipSystemStatsHCOutOctets	O	RO
ipSystemStatsInMcastPkts	O	RO
ipSystemStatsHCInMcastPkts	O	RO
ipSystemStatsInMcastOctets	O	RO
ipSystemStatsHCInMcastOctets	O	RO
ipSystemStatsOutMcastPkts	O	RO
ipSystemStatsHCOutMcastPkts	O	RO
ipSystemStatsOutMcastOctets	O	RO
ipSystemStatsHCOutMcastOctets	O	RO
ipSystemStatsInBcastPkts	O	RO
ipSystemStatsHCInBcastPkts	O	RO
ipSystemStatsOutBcastPkts	O	RO
ipSystemStatsHCOutBcastPkts	O	RO
ipSystemStatsDiscontinuityTime	O	RO
ipSystemStatsRefreshRate	O	RO

Object	CM	Access
ipIfStatsTableLastChange	O	RO
ipIfStatsTable **Note:** This table is required ONLY if routing is implemented.	O	N-Acc
ipIfStatsEntry	O	N-Acc
ipIfStatsIPVersion	O	N-Acc
ipIfStatsIfIndex	O	N-Acc
ipIfStatsInReceives	O	RO
ipIfStatsHCInReceives	O	RO
ipIfStatsInOctets	O	RO
ipIfStatsHCInOctets	O	RO
ipIfStatsInHdrErrors	O	RO
ipIfStatsInNoRoutes	O	RO
ipIfStatsInAddrErrors	O	RO
ipIfStatsInUnknownProtos	O	RO
ipIfStatsInTruncatedPkts	O	RO
ipIfStatsInForwDatagrams	O	RO
ipIfStatsHCInForwDatagrams	O	RO
ipIfStatsReasmReqds	O	RO
ipIfStatsReasmOKs	O	RO
ipIfStatsReasmFails	O	RO
ipIfStatsInDiscards	O	RO
ipIfStatsInDelivers	O	RO
ipIfStatsHCInDelivers	O	RO
ipIfStatsOutRequests	O	RO
ipIfStatsHCOutRequests	O	RO
ipIfStatsOutForwDatagrams	O	RO
ipIfStatsHCOutForwDatagrams	O	RO
ipIfStatsOutDiscards	O	RO
ipIfStatsOutFragReqds	O	RO
ipIfStatsOutFragOKs	O	RO
ipIfStatsOutFragFails	O	RO
ipIfStatsOutFragCreates	O	RO
ipIfStatsOutTransmits	O	RO
ipIfStatsHCOutTransmits	O	RO
ipIfStatsOutOctets	O	RO
ipIfStatsHCOutOctets	O	RO
ipIfStatsInMcastPkts	O	RO
ipIfStatsHCInMcastPkts	O	RO
ipIfStatsInMcastOctets	O	RO
ipIfStatsHCInMcastOctets	O	RO
ipIfStatsOutMcastPkts	O	RO
ipIfStatsHCOutMcastPkts	O	RO
ipIfStatsOutMcastOctets	O	RO
ipIfStatsHCOutMcastOctets	O	RO
ipIfStatsInBcastPkts	O	RO
ipIfStatsHCInBcastPkts	O	RO

ipIfStatsOutBcastPkts	O	RO
ipIfStatsHCOutBcastPkts	O	RO
ipIfStatsDiscontinuityTime	O	RO
ipIfStatsRefreshRate	O	RO
ipAddressPrefixTable **Note:** This table is required ONLY if routing is implemented.	O	N-Acc
ipAddressPrefixEntry	O	N-Acc
ipAddressPrefixIfIndex	O	N-Acc
ipAddressPrefixType	O	N-Acc
ipAddressPrefixPrefix	O	N-Acc
ipAddressPrefixLength	O	N-Acc
ipAddressPrefixOrigin	O	RO
ipAddressPrefixOnLinkFlag	O	RO
ipAddressPrefixAutonomousFlag	O	RO
ipAddressPrefixAdvPreferredLifetime	O	RO
ipAddressPrefixAdvValidLifetime	O	RO
Object	**CM**	**Access**
ipAddressSpinLock	O	RW
ipAddressTable	O	N-Acc
ipAddressEntry	O	N-Acc
ipAddressAddrType	O	N-Acc
ipAddressAddr	O	N-Acc
ipAddressIfIndex	O	RC
ipAddressType	O	RC
ipAddressPrefix	O	RO
ipAddressOrigin	O	RO
ipAddressStatus	O	RC
ipAddressCreated	O	RC
ipAddressLastChanged	O	RC
ipAddressRowStatus	O	RC
ipAddressStorageType	O	RC
ipNetToPhysicalTable **Note:** This table is required ONLY if routing is implemented.	O	N-Acc
ipNetToPhysicalEntry	O	N-Acc
ipNetToPhysicalIfIndex	O	N-Acc
ipNetToPhysicalNetAddressType	O	N-Acc
ipNetToPhysicalNetAddress	O	N-Acc
ipNetToPhysicalPhysAddress	O	RC
ipNetToPhysicalLastUpdated	O	RO
ipNetToPhysicalType	O	RC
ipNetToPhysicalState	O	RO
ipNetToPhysicalRowStatus	O	RC
ipDefaultRouterTable **Note:** This table is required ONLY if routing is implemented.	O	N-Acc
ipDefaultRouterEntry	O	N-Acc
ipDefaultRouterAddressType	O	N-Acc
ipDefaultRouterAddress	O	N-Acc

ipDefaultRouterIfIndex	O	N-Acc
ipDefaultRouterLifetime	O	RC
ipDefaultRouterPreference	O	RO
icmpStatsTable	M	N-Acc
icmpStatsEntry	M	N-Acc
icmpStatsIPVersion	M	N-Acc
icmpStatsInMsgs	M	RO
icmpStatsInErrors	M	RO
icmpStatsOutMsgs	M	RO
icmpStatsOutErrors	M	RO
icmpMsgStatsTable	M	N-Acc
icmpMsgStatsEntry	M	N-Acc
icmpMsgStatsIPVersion	M	N-Acc
icmpMsgStatsType	M	N-Acc
icmpMsgStatsInPkts	M	RO
icmpMsgStatsOutPkts	M	RO
UDP-MIB [RFC 4113]		
Object	**CM**	**Access**
UDPGroup		
udpInDatagrams	O	RO
udpNoPorts	O	RO
udpInErrors	O	RO
udpOutDatagrams	O	RO
udpEndpointTable	O	N-Acc
udpEndpointEntry	O	N-Acc
udpEndpointLocalAddressType	O	N-Acc
udpEndpointLocalAddress	O	N-Acc
udpEndpointLocalPort	O	N-Acc
udpEndpointRemoteAddressType	O	N-Acc
udpEndpointRemoteAddress	O	N-Acc
udpEndpointRemotePort	O	N-Acc
udpEndpointInstance	O	N-Acc
udpEndpointProcess	O	RO
TCP-MIB [RFC 4022]		
Object	**CM**	**Access**
tcpBaseGroup		
tcpRtoAlgorithm	O	RO
tcpRtoMin	O	RO
tcpRtoMax	O	RO
tcpMaxConn	O	RO
tcpActiveOpens	O	RO
tcpPassiveOpens	O	RO
tcpAttemptFails	O	RO
tcpEstabResets	O	RO
tcpCurrEstab	O	RO
tcpInSegs	O	RO
tcpOutSegs	O	RO

tcpRetransSegs	O	RO
tcpInErrs	O	RO
tcpOutRsts	O	RO
tcpHCGroup		
tcpHCInSegs	O	RO
tcpHCOutSegs	O	RO
tcpConnectionTable	O	N-Acc
tcpConnectionEntry	O	N-Acc
tcpConnectionLocalAddressType	O	N-Acc
tcpConnectionLocalAddress	O	N-Acc
tcpConnectionLocalPort	O	N-Acc
tcpConnectionRemAddressType	O	N-Acc
tcpConnectionRemAddress	O	N-Acc
tcpConnectionRemPort	O	N-Acc
tcpConnectionState	O	RW
tcpConnectionProcess	O	RO
tcpListenerTable	O	N-Acc
tcpListenerEntry	O	N-Acc
tcpListenerLocalAddressType	O	N-Acc
tcpListenerLocalAddress	O	N-Acc
tcpListenerLocalPort	O	N-Acc
tcpListenerProcess	O	RO
SNMPv2-MIB [RFC 3418]		
Object	**CM**	**Access**
SystemGroup		
sysDescr	M	RO
sysObjectID	M	RO
sysUpTime	M	RO
sysContact	M	RW
sysName	M	RW
sysLocation	M	RW
sysServices	M	RO
sysORLastChange	M	RO
sysORTable	M	N-Acc
sysOREntry	M	N-Acc
sysORIndex	M	N-Acc
sysORID	M	RO
sysORDescr	M	RO
sysORUpTime	M	RO
SNMPGroup		
snmpInPkts	M	RO
snmpInBadVersions	M	RO
snmpOutPkts	Ob	RO
snmpInBadCommunityNames	M	RO
snmpInBadCommunityUses	M	RO
snmpInASNParseErrs	M	RO
snmpInTooBigs	Ob	RO

snmpInNoSuchNames	Ob	RO
snmpInBadValues	Ob	RO
snmpInReadOnlys	Ob	RO
snmpInGenErrs	Ob	RO
snmpInTotalReqVars	Ob	RO
snmpInTotalSetVars	Ob	RO
snmpInGetRequests	Ob	RO
snmpInGetNexts	Ob	RO
snmpInSetRequests	Ob	RO
snmpInGetResponses	Ob	RO
snmpInTraps	Ob	RO
snmpOutTooBigs	Ob	RO
snmpOutNoSuchNames	Ob	RO
snmpOutBadValues	Ob	RO
snmpOutGenErrs	Ob	RO
snmpOutGetRequests	Ob	RO
snmpOutGetNexts	Ob	RO
snmpOutSetRequests	Ob	RO
snmpOutGetResponses	Ob	RO
snmpOutTraps	Ob	RO
snmpEnableAuthenTraps	M	RW
snmpSilentDrops	M	RO
snmpProxyDrops	M	RO
snmpTrapsGroup		
coldStart	O	Acc-FN
warmStart	O	Acc-FN
authenticationFailure	M	Acc-FN
snmpSetGroup		
snmpSetSerialNo	M	RW
Etherlike-MIB [RFC 3635]		
Object	**CM**	**Access**
dot3StatsTable	O	N-Acc
dot3StatsEntry	O	N-Acc
dot3StatsIndex	O	RO
dot3StatsAlignmentErrors	O	RO
dot3StatsFCSErrors	O	RO
dot3StatsInternalMacTransmitErrors	O	RO
dot3StatsFrameTooLongs	O	RO
dot3StatsInternalMacReceiveErrors	O	RO
dot3StatsSymbolErrors	O	RO
dot3StatsSingleCollisionFrames	O	RO
dot3StatsMultipleCollisionFrames	O	RO
dot3StatsDeferredTransmissions	O	RO
dot3StatsLateCollisions	O	RO
dot3StatsExcessiveCollisions	O	RO
dot3StatsCarrierSenseErrors	O	RO
dot3StatsDuplexStatus	O	RO

dot3StatsSQETestErrors	O	RO
dot3CollTable	O	N-Acc
dot3CollEntry	O	N-Acc
dot3CollCount	O	NA
dot3CollFrequencies	O	RO
dot3ControlTable	O	N-Acc
dot3ControlEntry	O	N-Acc
dot3ControlFunctionsSupported	O	RO
dot3ControlInUnknownOpcodes	O	RO
dot3PauseTable	O	N-Acc
dot3PauseEntry	O	N-Acc
dot3PauseAdminMode	O	RW
dot3PauseOperMode	O	RO
dot3InPauseFrames	O	RO
dot3OutPauseFrames	O	RO
DOCS-IETF-BPI2-MIB [RFC 4131]		
Object	**CM**	**Access**
docsBpi2CmBaseTable	M	N-Acc
docsBpi2CmBaseEntry	M	N-Acc
docsBpi2CmPrivacyEnable	M	RO
docsBpi2CmPublicKey	M	RO
docsBpi2CmAuthState	M	RO
docsBpi2CmAuthKeySequenceNumber	M	RO
docsBpi2CmAuthExpiresOld	M	RO
docsBpi2CmAuthExpiresNew	M	RO
docsBpi2CmAuthReset	M	RW
docsBpi2CmAuthGraceTime	M	RO
docsBpi2CmTEKGraceTime	M	RO
docsBpi2CmAuthWaitTimeout	M	RO
docsBpi2CmReauthWaitTimeout	M	RO
docsBpi2CmOpWaitTimeout	M	RO
docsBpi2CmRekeyWaitTimeout	M	RO
docsBpi2CmAuthRejectWaitTimeout	M	RO
docsBpi2CmSAMapWaitTimeout	M	RO
docsBpi2CmSAMapMaxRetries	M	RO
docsBpi2CmAuthentInfos	M	RO
docsBpi2CmAuthRequests	M	RO
docsBpi2CmAuthReplies	M	RO
docsBpi2CmAuthRejects	M	RO
docsBpi2CmAuthInvalids	M	RO
docsBpi2CmAuthRejectErrorCode	M	RO
docsBpi2CmAuthRejectErrorString	M	RO
docsBpi2CmAuthInvalidErrorCode	M	RO
docsBpi2CmAuthInvalidErrorString	M	RO
docsBpi2CmTEKTable	M	N-Acc
docsBpi2CmTEKEntry	M	N-Acc
docsBpi2CmTEKSAId	M	N-Acc

docsBpi2CmTEKSAType	M	RO
docsBpi2CmTEKDataEncryptAlg	M	RO
docsBpi2CmTEKDataAuthentAlg	M	RO
docsBpi2CmTEKState	M	RO
docsBpi2CmTEKKeySequenceNumber	M	RO
docsBpi2CmTEKExpiresOld	M	RO
docsBpi2CmTEKExpiresNew	M	RO
docsBpi2CmTEKKeyRequests	M	RO
docsBpi2CmTEKKeyReplies	M	RO
docsBpi2CmTEKKeyRejects	M	RO
docsBpi2CmTEKInvalids	M	RO
docsBpi2CmTEKAuthPends	M	RO
docsBpi2CmTEKKeyRejectErrorCode	M	RO
docsBpi2CmTEKKeyRejectErrorString	M	RO
docsBpi2CmTEKInvalidErrorCode	M	RO
docsBpi2CmTEKInvalidErrorString	M	RO
docsBpi2CmIpMulticastMapTable	D	N-Acc
docsBpi2CmIpMulticastMapEntry	D	N-Acc
docsBpi2CmIpMulticastIndex	D	N-Acc
docsBpi2CmIpMulticastAddressType	D	RO
docsBpi2CmIpMulticastAddress	D	RO
docsBpi2CmIpMulticastSAId	D	RO
docsBpi2CmIpMulticastSAMapState	D	RO
docsBpi2CmIpMulticastSAMapRequests	D	RO
docsBpi2CmIpMulticastSAMapReplies	D	RO
docsBpi2CmIpMulticastSAMapRejects	D	RO
docsBpi2CmIpMulticastSAMapRejectErrorCode	D	RO
docsBpi2CmIpMulticastSAMapRejectErrorString	D	RO
Object	**CM**	**Access**
docsBpi2CmDeviceCertTable	M	N-Acc
docsBpi2CmDeviceCertEntry	M	N-Acc
docsBpi2CmDeviceCmCert	M	RW/RO
docsBpi2CmDeviceManufCert	M	RO
docsBpi2CmCryptoSuiteTable	M	N-Acc
docsBpi2CmCryptoSuiteEntry	M	N-Acc
docsBpi2CmCryptoSuiteIndex	M	N-Acc
docsBpi2CmCryptoSuiteDataEncryptAlg	M	RO
docsBpi2CmCryptoSuiteDataAuthentAlg	M	RO
docsBpi2CodeDownloadGroup		
docsBpi2CodeDownloadStatusCode	M	RO
docsBpi2CodeDownloadStatusString	M	RO
docsBpi2CodeMfgOrgName	M	RO
docsBpi2CodeMfgCodeAccessStart	M	RO
docsBpi2CodeMfgCvcAccessStart	M	RO
docsBpi2CodeCoSignerOrgName	M	RO
docsBpi2CodeCoSignerCodeAccessStart	M	RO
docsBpi2CodeCoSignerCvcAccessStart	M	RO

docsBpi2CodeCvcUpdate	M	RW
DOCS-IFEXT2-MIB		
Object	**CM**	**Access**
docsIfExt2CmMscStatusTable	M	N-Acc
docsIfExt2CmMscStatusEntry	M	N-Acc
docsIfExt2CmMscStatusState	M	RO
docsIfExt2CmMscStatusPowerShortfall	M	RO
docsIfExt2CmMscStatusCodeRatio	M	RO
docsIfExt2CmMscStatusMaximumScheduledCodes	M	RO
docsIfExt2CmMscStatusPowerHeadroom	M	RO
docsIfExt2CmMscStatusEffectivePower	M	RO
docsIfExt2CmMscStatusIUC2Control	M	RW
docsIfExt2CmClearLearnedMacAddresses	M	RW
HOST-RESOURCES-MIB [RFC 2790]		
Object	**CM**	**Access**
hrDeviceTable	O	N-Acc
hrDeviceEntry	O	N-Acc
hrDeviceIndex	O	RO
hrDeviceType	O	RO
hrDeviceDescr	O	RO
hrDeviceID	O	RO
hrDeviceStatus	O	RO
hrDeviceErrors	O	RO
hrSystem		
hrMemorySize	O	RO
hrStorageTable	O	N-Acc
hrStorageEntry	O	N-Acc
hrStorageIndex	O	RO
hrStorageType	O	RO
hrStorageDescr	O	RO
hrStorageAllocationUnits	O	RO
hrStorageSize	O	RO
hrStorageUsed	O	RO
hrStorageAllocationFailures	O	RO
hrSWRunTable	O	N-Acc
hrSWRunEntry	O	N-Acc
hrSWRunIndex	O	RO
hrSWRunName	O	RO
hrSWRunID	O	RO
hrSWRunPath	O	RO
hrSWRunParameters	O	RO
hrSWRunType	O	RO
hrSWRunStatus	O	RO
hrSWRunPerfTable	O	N-Acc
hrSWRunPerfEntry	O	N-Acc
hrSWRunPerfCPU	O	RO
hrSWRunPerfMem	O	RO

hrProcessorTable	O	N-Acc
hrProcessorEntry	O	N-Acc
hrProcessorFrwID	O	RO
hrProcessorLoad	O	RO
ENTITY-MIB [RFC 6933]		
Object	**CM**	**Access**
entPhysicalTable	O	N-Acc
entPhysicalEntry	O	N-Acc
entPhysicalIndex	O	RO
entPhysicalDescr	O	RO
entPhysicalVendorType	O	RO
entPhysicalContainedIn	O	RO
entPhysicalClass	O	RO
entPhysicalParentRelPos	O	RO
entPhysicalName	O	RO
entPhysicalHardwareRev	O	RO
entPhysicalFirmwareRev	O	RO
entPhysicalSoftwareRev	O	RO
entPhysicalSerialNum	O	RO
entPhysicalMfgName	O	RO
entPhysicalModelName	O	RO
entPhysicalAlias	O	RO
entPhysicalAssetID	O	RO
entPhysicalIsFRU	O	RO
entPhysicalMfgDate	O	RO
entPhysicalUris	O	RO
entPhysicalUUID	O	RO
ENTITY-SENSOR-MIB [RFC 3433]		
Object	**CM**	**Access**
entPhySensorTable	O	N-Acc
entPhySensorEntry	O	N-Acc
entPhySensorType	O	RO
entPhySensorScale	O	RO
entPhySensorPrecision	O	RO
entPhySensorValue	O	RO
entPhySensorOperStatus	O	RO
entPhySensorUnitsDisplay	O	RO
entPhySensorValueTimeStamp	O	RO
entPhySensorValueUpdateRate	O	RO

SNMP-USM-DH-OBJECTS-MIB [RFC 2786]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
usmDHParameters	N-Sup		usmDHParameters	M	RW
usmDHUserKeyTable	N-Sup		**usmDHUserKeyTable**	M	N-Acc

usmDHUserKeyEntry	N-Sup		**usmDHUserKeyEntry**	M	N-Acc
usmDHUserAuthKeyChange	N-Sup		usmDHUserAuthKeyChange	M	RC
usmDHUserOwnAuthKeyChange	N-Sup		usmDHUserOwnAuthKeyChange	M	RC
usmDHUserPrivKeyChange	N-Sup		usmDHUserPrivKeyChange	M	RC
usmDHUserOwnPrivKeyChange	N-Sup		usmDHUserOwnPrivKeyChange	M	RC
usmDHKickstartTable	N-Sup		**usmDHKickstartTable**	M	N-Acc
usmDHKickstartEntry	N-Sup		**usmDHKickstartEntry**	M	N-Acc
usmDHKickstartIndex	N-Sup		usmDHKickstartIndex	M	N-Acc
usmDHKickstartMyPublic	N-Sup		usmDHKickstartMyPublic	M	RO
usmDHKickstartMgrPublic	N-Sup		usmDHKickstartMgrPublic	M	RO
usmDHKickstartSecurityName	N-Sup		usmDHKickstartSecurityName	M	RO
SNMP-VIEW-BASED-ACM-MIB [RFC 3415]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
vacmContextTable	N-Sup		**vacmContextTable**	M	N-Acc
vacmContextEntry	N-Sup		**vacmContextEntry**	M	N-Acc
vacmContextName	N-Sup		vacmContextName	M	RO
vacmSecurityToGroupTable	N-Sup		**vacmSecurityToGroupTable**	M	N-Acc
vacmSecurityToGroupEntry	N-Sup		**vacmSecurityToGroupEntry**	M	N-Acc
vacmSecurityModel	N-Sup		vacmSecurityModel	M	N-Acc
vacmSecurityName	N-Sup		vacmSecurityName	M	N-Acc
vacmGroupName	N-Sup		vacmGroupName	M	RC
vacmSecurityToGroupStorageType	N-Sup		vacmSecurityToGroupStorageType	M	RC
vacmSecurityToGroupStatus	N-Sup		vacmSecurityToGroupStatus	M	RC
vacmAccessTable	N-Sup		**vacmAccessTable**	M	N-Acc
vacmAccessEntry	N-Sup		**vacmAccessEntry**	M	N-Acc
vacmAccessContextPrefix	N-Sup		vacmAccessContextPrefix	M	N-Acc
vacmAccessSecurityModel	N-Sup		vacmAccessSecurityModel	M	N-Acc
vacmAccessSecurityLevel	N-Sup		vacmAccessSecurityLevel	M	N-Acc
vacmAccessContextMatch	N-Sup		vacmAccessContextMatch	M	RC
vacmAccessReadViewName	N-Sup		vacmAccessReadViewName	M	RC
vacmAccessWriteViewName	N-Sup		vacmAccessWriteViewName	M	RC
vacmAccessNotifyViewName	N-Sup		vacmAccessNotifyViewName	M	RC
vacmAccessStorageType	N-Sup		vacmAccessStorageType	M	RC
vacmAccessStatus	N-Sup		vacmAccessStatus	M	RC
vacmViewSpinLock	N-Sup		vacmViewSpinLock	M	RW
vacmViewTreeFamilyTable	N-Sup		**vacmViewTreeFamilyTable**	M	N-Acc
vacmViewTreeFamilyEntry	N-Sup		**vacmViewTreeFamilyEntry**	M	N-Acc
vacmViewTreeFamilyViewName	N-Sup		vacmViewTreeFamilyViewName	M	N-Acc
vacmViewTreeFamilySubtree	N-Sup		vacmViewTreeFamilySubtree	M	N-Acc
vacmViewTreeFamilyMask	N-Sup		vacmViewTreeFamilyMask	M	RC
vacmViewTreeFamilyType	N-Sup		vacmViewTreeFamilyType	M	RC
vacmViewTreeFamilyStorageType	N-Sup		vacmViewTreeFamilyStorageType	M	RC
vacmViewTreeFamilyStatus	N-Sup		vacmViewTreeFamilyStatus	M	RC

SNMP-COMMUNITY-MIB [RFC 3584]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
snmpCommunityTable	N-Sup		**snmpCommunityTable**	M	N-Acc
snmpCommunityEntry	N-Sup		**snmpCommunityEntry**	M	N-Acc
snmpCommunityIndex	N-Sup		snmpCommunityIndex	M	N-Acc
snmpCommunityName	N-Sup		snmpCommunityName	M	RC
snmpCommunitySecurityName	N-Sup		snmpCommunitySecurityName	M	RC
snmpCommunityContextEngineID	N-Sup		snmpCommunityContextEngineID	M	RC
snmpCommunityContextName	N-Sup		snmpCommunityContextName	M	RC
snmpCommunityTransportTag	N-Sup		snmpCommunityTransportTag	M	RC
snmpCommunityStorageType	N-Sup		snmpCommunityStorageType	M	RC
snmpCommunityStatus	N-Sup		snmpCommunityStatus	M	RC
snmpTargetAddrExtTable	N-Sup		**snmpTargetAddrExtTable**	M	N-Acc
snmpTargetAddrExtEntry	N-Sup		**snmpTargetAddrExtEntry**	M	N-Acc
snmpTargetAddrTMask	N-Sup		snmpTargetAddrTMask	M	RC
snmpTargetAddrMMS	N-Sup		snmpTargetAddrMMS	M	RC
snmpTrapAddress	N-Sup		snmpTrapAddress	O	ACC-FN
snmpTrapCommunity	N-Sup		snmpTrapCommunity	O	ACC-FN
SNMP-FRAMEWORK-MIB [RFC 3411]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
snmpEngineGroup			**snmpEngineGroup**		
snmpEngineID	N-Sup		snmpEngineID	M	RO
snmpEngineBoots	N-Sup		snmpEngineBoots	M	RO
snmpEngineTime	N-Sup		snmpEngineTime	M	RO
snmpEngineMaxMessageSize	N-Sup		snmpEngineMaxMessageSize	M	RO
SNMP-MPD-MIB [RFC 3412]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
snmpMPDStats			**snmpMPDStats**		
snmpUnknownSecurityModels	N-Sup		snmpUnknownSecurityModels	M	RO
snmpInvalidMsgs	N-Sup		snmpInvalidMsgs	M	RO
snmpUnknownPDUHandlers	N-Sup		snmpUnknownPDUHandlers	M	RO
SNMP-TARGET-MIB [RFC 3413]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
snmpTargetSpinLock	N-Sup		snmpTargetSpinLock	M	RW
snmpTargetAddrTable	N-Sup		**snmpTargetAddrTable**	M	N-Acc
snmpTargetAddrEntry	N-Sup		**snmpTargetAddrEntry**	M	N-Acc
snmpTargetAddrName	N-Sup		snmpTargetAddrName	M	N-Acc
snmpTargetAddrTDomain	N-Sup		snmpTargetAddrTDomain	M	RC
snmpTargetAddrTAddress	N-Sup		snmpTargetAddrTAddress	M	RC
snmpTargetAddrTimeout	N-Sup		snmpTargetAddrTimeout	M	RC

snmpTargetAddrRetryCount	N-Sup		snmpTargetAddrRetryCount	M	RC
snmpTargetAddrTagList	N-Sup		snmpTargetAddrTagList	M	RC
snmpTargetAddrParams	N-Sup		snmpTargetAddrParams	M	RC
snmpTargetAddrStorageType	N-Sup		snmpTargetAddrStorageType	M	RC
snmpTargetAddrRowStatus	N-Sup		snmpTargetAddrRowStatus	M	RC
snmpTargetParamsTable	N-Sup		**snmpTargetParamsTable**	M	N-Acc
snmpTargetParamsEntry	N-Sup		**snmpTargetParamsEntry**	M	N-Acc
snmpTargetParamsName	N-Sup		snmpTargetParamsName	M	N-Acc
snmpTargetParamsMPModel	N-Sup		snmpTargetParamsMPModel	M	RC
snmpTargetParamsSecurityModel	N-Sup		snmpTargetParamsSecurityModel	M	RC
snmpTargetParamsSecurityName	N-Sup		snmpTargetParamsSecurityName	M	RC
snmpTargetParamsSecurityLevel	N-Sup		snmpTargetParamsSecurityLevel	M	RC
snmpTargetParamsStorageType	N-Sup		snmpTargetParamsStorageType	M	RC
snmpTargetParamsRowStatus	N-Sup		snmpTargetParamsRowStatus	M	RC
snmpUnavailableContexts	N-Sup		snmpUnavailableContexts	M	RO
snmpUnknownContexts	N-Sup		snmpUnknownContexts	M	RO
snmpNotifyTable	N-Sup		**snmpNotifyTable**	M	N-Acc
snmpNotifyEntry	N-Sup		**snmpNotifyEntry**	M	N-Acc
snmpNotifyName	N-Sup		snmpNotifyName	M	N-Acc
snmpNotifyTag	N-Sup		snmpNotifyTag	M	RC
snmpNotifyType	N-Sup		snmpNotifyType	M	RC
snmpNotifyStorageType	N-Sup		snmpNotifyStorageType	M	RC
snmpNotifyRowStatus	N-Sup		snmpNotifyRowStatus	M	RC
snmpNotifyFilterProfileTable	N-Sup		snmpNotifyFilterProfileTable	M	N-Acc
snmpNotifyFilterProfileEntry	N-Sup		snmpNotifyFilterProfileEntry	M	N-Acc
snmpNotifyFilterProfileName	N-Sup		snmpNotifyFilterProfileName	M	RC
snmpNotifyFilterProfileStorType	N-Sup		snmpNotifyFilterProfileStorType	M	RC
snmpNotifyFilterProfileRowStatus	N-Sup		snmpNotifyFilterProfileRowStatus	M	RC
snmpNotifyFilterTable	N-Sup		**snmpNotifyFilterTable**	M	N-Acc
snmpNotifyFilterEntry	N-Sup		**snmpNotifyFilterEntry**	M	N-Acc
snmpNotifyFilterSubtree	N-Sup		snmpNotifyFilterSubtree	M	N-Acc
snmpNotifyFilterMask	N-Sup		snmpNotifyFilterMask	M	RC
snmpNotifyFilterType	N-Sup		snmpNotifyFilterType	M	RC
snmpNotifyFilterStorageType	N-Sup		snmpNotifyFilterStorageType	M	RC
snmpNotifyFilterRowStatus	N-Sup		snmpNotifyFilterRowStatus	M	RC
SNMP-USER-BASED-SM-MIB [RFC 3414]					
Object	**CM in NmAccess Mode**	**Access**		**CM in SNMP Coexistence Mode**	**Access**
usmStats			usmStats		
usmStatsUnsupportedSecLevels	N-Sup		usmStatsUnsupportedSecLevels	M	RO
usmStatsNotInTimeWindows	N-Sup		usmStatsNotInTimeWindows	M	RO
usmStatsUnknownUserNames	N-Sup		usmStatsUnknownUserNames	M	RO
usmStatsUnknownEngineIDs	N-Sup		usmStatsUnknownEngineIDs	M	RO
usmStatsWrongDigests	N-Sup		usmStatsWrongDigests	M	RO
usmStatsDecryptionErrors	N-Sup		usmStatsDecryptionErrors	M	RO
usmUser			usmUser		

usmUserSpinLock	N-Sup		usmUserSpinLock	M	RW
usmUserTable	N-Sup		**usmUserTable**	M	N-Acc
usmUserEntry	N-Sup		**usmUserEntry**	M	N-Acc
usmUserEngineID	N-Sup		usmUserEngineID	M	N-Acc
usmUserName	N-Sup		usmUserName	M	N-Acc
usmUserSecurityName	N-Sup		usmUserSecurityName	M	RO
usmUserCloneFrom	N-Sup		usmUserCloneFrom	M	RC
usmUserAuthProtocol	N-Sup		usmUserAuthProtocol	M	RC
usmUserAuthKeyChange	N-Sup		usmUserAuthKeyChange	M	RC
usmUserOwnAuthKeyChange	N-Sup		usmUserOwnAuthKeyChange	M	RC
usmUserPrivProtocol	N-Sup		usmUserPrivProtocol	M	RC
usmUserPrivKeyChange	N-Sup		usmUserPrivKeyChange	M	RC
usmUserOwnPrivKeyChange	N-Sup		usmUserOwnPrivKeyChange	M	RC
usmUserPublic	N-Sup		usmUserPublic	M	RC
usmUserStorageType	N-Sup		usmUserStorageType	M	RC
usmUserStatus	N-Sup		usmUserStatus	M	RC

IGMP-STD-MIB [RFC 2933]		
Object	**CM**	**Access**
igmpInterfaceTable	D	N-Acc
igmpInterfaceEntry	D	N-Acc
igmpInterfaceIfIndex	D	N-Acc
igmpInterfaceQueryInterval	D	RC
igmpInterfaceStatus	D	RC
igmpInterfaceVersion	D	RO
igmpInterfaceQuerier	D	RO
igmpInterfaceQueryMaxResponseTime	D	RC
igmpInterfaceQuerierUpTime	D	RO
igmpInterfaceQuerierExpiryTime	D	RO
igmpInterfaceVersion1QuerierTimer	D	RO
igmpInterfaceWrongVersionQueries	D	RO
igmpInterfaceJoins	D	RO
igmpInterfaceProxyIfIndex	D	RO
igmpInterfaceGroups	D	RO
igmpInterfaceRobustness	D	RC
igmpInterfaceLastMembQueryIntvl	D	RC
igmpCacheTable	D	N-Acc
igmpCacheEntry	D	N-Acc
igmpCacheAddress	D	N-Acc
igmpCacheIfIndex	D	N-Acc
igmpCacheSelf	D	RC
igmpCacheLastReporter	D	RO
igmpCacheUpTime	D	RO
igmpCacheExpiryTime	D	RO
igmpCacheStatus	D	RO
igmpCacheVersion1HostTimer	D	RO

[DOCS-QOS3-MIB]		
Object	**CM**	**Access**
docsQosPktClassTable	M	N-Acc
docsQosPktClassEntry	M	N-Acc
docsQosPktClassId	M	N-Acc
docsQosPktClassDirection	M	RO
docsQosPktClassPriority	M	RO
docsQosPktClassIpTosLow	M	RO
docsQosPktClassIpTosHigh	M	RO
docsQosPktClassIpTosMask	M	RO
docsQosPktClassIpProtocol	M	RO
docsQosPktClassIpSourceAddr	M	RO
docsQosPktClassIpSourceMask	M	RO
docsQosPktClassIpDestAddr	M	RO
docsQosPktClassIpDestMask	M	RO
docsQosPktClassSourcePortStart	M	RO
docsQosPktClassSourcePortEnd	M	RO
docsQosPktClassDestPortStart	M	RO
docsQosPktClassDestPortEnd	M	RO
docsQosPktClassDestMacAddr	M	RO
docsQosPktClassDestMacMask	M	RO
docsQosPktClassSourceMacAddr	M	RO
docsQosPktClassEnetProtocolType	M	RO
docsQosPktClassEnetProtocol	M	RO
docsQosPktClassUserPriLow	M	RO
docsQosPktClassUserPriHigh	M	RO
docsQosPktClassVlanId	M	RO
docsQosPktClassState	M	RO
docsQosPktClassPkts	M	RO
docsQosPktClassBitMap	M	RO
docsQosPktClassIpAddrType	M	RO
docsQosPktClassFlowLabel	M	RO
docsQosPktClassIcmpTypeHigh	M	RO
docsQosPktClassIcmpTypeLow	M	RO
docsQosPktClassCmInterfaceMask	M	RO
docsQosParamSetTable	M	N-Acc
docsQosParamSetEntry	M	N-Acc
docsQosParamSetServiceClassName	M	RO
docsQosParamSetPriority	M	RO
docsQosParamSetMaxTrafficRate	M	RO
docsQosParamSetMaxTrafficBurst	M	RO
docsQosParamSetMinReservedRate	M	RO
docsQosParamSetMinReservedPkt	M	RO
docsQosParamSetActiveTimeout	M	RO
docsQosParamSetAdmittedTimeout	M	RO
docsQosParamSetMaxConcatBurst	M	RO
docsQosParamSetSchedulingType	M	RO

docsQosParamSetNomPollInterval	M	RO
docsQosParamSetTolPollJitter	M	RO
docsQosParamSetUnsolicitGrantSize	M	RO
docsQosParamSetNomGrantInterval	M	RO
docsQosParamSetTolGrantJitter	M	RO
docsQosParamSetGrantsPerInterval	M	RO
docsQosParamSetTosAndMask	M	RO
docsQosParamSetTosOrMask	M	RO
docsQosParamSetMaxLatency	M	RO
docsQosParamSetType	M	N-Acc
docsQosParamSetRequestPolicyOct	M	RO
docsQosParamSetBitMap	M	RO
docsQosParamSetServiceFlowId	M	N-Acc
docsQosParamSetRequiredAttrMask	M	RO
docsQosParamSetForbiddenAttrMask	M	RO
docsQosParamSetAttrAggrRuleMask	M	RO
docsQosParamSetAppId	M	RO
docsQosParamSetMultiplierContentionReqWindow	M	RO
docsQosParamSetMultiplierBytesReq	M	RO
docsQosParamSetMaxReqPerSidCluster	D	RO
docsQosParamSetMaxOutstandingBytesPerSidCluster	D	RO
docsQosParamSetMaxTotBytesReqPerSidCluster	D	RO
docsQosParamSetMaxTimeInSidCluster	D	RO
docsQosParamSetPeakTrafficRate	M	RO
docsQosParamSetDsResequencing	M	RO
docsQosParamSetMinimumBuffer	M	RO
docsQosParamSetTargetBuffer	M	RO
docsQosParamSetMaximumBuffer	M	RO
docsQosParamSetAqmDisabled	M	RO
docsQosParamSetAqmLatencyTarget	M	RO
docsQosParamSetHCMaxTrafficRate	M	RO
docsQosParamSetHCMinReservedRate	M	RO
docsQosParamSetHCPeakTrafficRate	M	RO
docsQosParamSetAqmAlgInUse	M	RO
docsQosParamSetGuaranteedGrantInterval	M	RO
docsQosParamSetGuaranteedGrantRate	M	RO
docsQosParamSetGuaranteedRequestInterval	M	RO
docsQosParamSetImmedAqmMinThreshold	M	RO
docsQosParamSetImmedAqmRangeExponentRampFunc	M	RO
docsQosParamSetDataRateUnitSetting	M	RO
docsQosServiceFlowTable	M	N-Acc
docsQosServiceFlowEntry	M	N-Acc
docsQosServiceFlowId	M	N-Acc
docsQosServiceFlowSID	M	RO
docsQosServiceFlowDirection	M	RO
docsQosServiceFlowPrimary	M	RO
docsQosServiceFlowParamSetTypeStatus	M	RO

docsQosServiceFlowChSetId	M	RO
docsQosServiceFlowAttrAssignSuccess	M	RO
docsQosServiceFlowDsid	M	RO
docsQosServiceFlowMaxReqPerSidCluster	M	RO
docsQosServiceFlowMaxOutstandingBytesPerSidCluster	M	RO
docsQosServiceFlowMaxTotBytesReqPerSidCluster	M	RO
docsQosServiceFlowMaxTimeInSidCluster	M	RO
docsQosServiceFlowBufferSize	M	RO
docsQosServiceFlowStatsTable	M	N-Acc
docsQosServiceFlowStatsEntry	M	N-Acc
docsQosServiceFlowPkts	M	RO
docsQosServiceFlowOctets	M	RO
docsQosServiceFlowTimeCreated	M	RO
docsQosServiceFlowTimeActive	M	RO
docsQosServiceFlowPolicedDropPkts	M	RO
docsQosServiceFlowPolicedDelayPkts	M	RO
docsQosServiceFlowAqmDroppedPkts	M	RO
docsQosDynamicServiceStatsTable	M	N-Acc
docsQosDynamicServiceStatsEntry	M	N-Acc
docsQosIfDirection	M	N-Acc
docsQosDSAReqs	M	RO
docsQosDSARsps	M	RO
docsQosDSAAcks	M	RO
docsQosDSCReqs	M	RO
docsQosDSCRsps	M	RO
docsQosDSCAcks	M	RO
docsQosDSDReqs	M	RO
docsQosDSDRsps	M	RO
docsQosDynamicAdds	M	RO
docsQosDynamicAddFails	M	RO
docsQosDynamicChanges	M	RO
docsQosDynamicChangeFails	M	RO
docsQosDynamicDeletes	M	RO
docsQosDynamicDeleteFails	M	RO
docsQosDCCReqs	M	RO
docsQosDCCRsps	M	RO
docsQosDCCAcks	M	RO
docsQosDCCs	M	RO
docsQosDCCFails	M	RO
docsQosDCCRspDeparts	M	RO
docsQosDCCRspArrives	M	RO
docsQosDbcReqs	M	RO
docsQosDbcRsps	M	RO
docsQosDbcAcks	M	RO
docsQosDbcSuccesses	M	RO
docsQosDbcFails	M	RO
docsQosDbcPartial	M	RO

docsQosServiceFlowSidClusterTable	M	N-Acc
docsQosServiceFlowSidClusterEntry	M	N-Acc
docsQosServiceFlowSidClusterId	M	N-Acc
docsQosServiceFlowSidClusterUcid	M	N-Acc
docsQosServiceFlowSidClusterSid	M	RO
docsQosAggregateServiceFlowTable	M	N-Acc
docsQosAggregateServiceFlowEntry	M	N-Acc
docsQosAggregateServiceFlowId	M	N-Acc
docsQosAggregateServiceFlowDirection	M	RO
docsQosAggregateServiceFlowPriority	M	RO
docsQosAggregateServiceFlowMaxAggregateTrafficRate	M	RO
docsQosAggregateServiceFlowMaxTrafficBurst	M	RO
docsQosAggregateServiceFlowMinReservedRate	M	RO
docsQosAggregateServiceFlowMinReservedPkt	M	RO
docsQosAggregateServiceFlowPeakTrafficRate	M	RO
docsQosAggregateServiceFlowDataRateUnitSetting	M	RO
docsQosAggregateServiceFlowLowLatencyAsf	M	RO
docsQosAggregateServiceFlowLowLatencySfId	M	RO
docsQosAggregateServiceFlowClassicSfScn	M	RO
docsQosAggregateServiceFlowLatencySfScn	M	RO
docsQosAggregateServiceFlowAqmCouplingFactor	M	RO
docsQosAggregateServiceFlowSchedulingWeight	M	RO
docsQosAggregateServiceFlowQpEnable	M	RO
docsQosAggregateServiceFlowQpLatencyThreshold	M	RO
docsQosAggregateServiceFlowQpQueuingScoreThreshold	M	RO
docsQosAggregateServiceFlowQpDrainRateExponent	M	RO
docsQosAggregateServiceFlowHCMaxAggregateTrafficRate	M	RO
docsQosAggregateServiceFlowHCMinReservedRate	M	RO
docsQosAggregateServiceFlowHCPeakTrafficRate	M	RO
docsQosAggregateServiceFlowStatsTable	M	N-Acc
docsQosAggregateServiceFlowStatsEntry	M	N-Acc
docsQosAggregateServiceFlowStatsPkts	M	RO
docsQosAggregateServiceFlowStatsOctets	M	RO
docsQosAggregateServiceFlowStatsTimeCreated	M	RO
docsQosAggregateServiceFlowStatsTimeActive	M	RO
docsQosSfLatencyHistCfgTable	**M**	**N-Acc**
docsQosSfLatencyHistCfgEntry	**M**	**N-Acc**
docsQosSfLatencyHistCfgStatus	M	RC
docsQosSfLatencySfLabel	M	RC
docsQosSfLatencyBin1UpperEdge	M	RC
docsQosSfLatencyBin2UpperEdge	M	RC
docsQosSfLatencyBin3UpperEdge	M	RC
docsQosSfLatencyBin4UpperEdge	M	RC
docsQosSfLatencyBin5UpperEdge	M	RC
docsQosSfLatencyBin6UpperEdge	M	RC
docsQosSfLatencyBin7UpperEdge	M	RC
docsQosSfLatencyBin8UpperEdge	M	RC

docsQosSfLatencyBin9UpperEdge	M	RC
docsQosSfLatencyBin10UpperEdge	M	RC
docsQosSfLatencyBin11UpperEdge	M	RC
docsQosSfLatencyBin12UpperEdge	M	RC
docsQosSfLatencyBin13UpperEdge	M	RC
docsQosSfLatencyBin14UpperEdge	M	RC
docsQosSfLatencyBin15UpperEdge	M	RC
docsQosSfLatencyStatsTable	M	N-Acc
docsQosSfLatencyStatsEntry	M	N-Acc
docsQosSfLatencyMaxLatency	M	RO
docsQosSfLatencyNumHistUpdates	M	RO
docsQosSfLatencyBin1Pkts	M	RO
docsQosSfLatencyBin2Pkts	M	RO
docsQosSfLatencyBin3Pkts	M	RO
docsQosSfLatencyBin4Pkts	M	RO
docsQosSfLatencyBin5Pkts	M	RO
docsQosSfLatencyBin6Pkts	M	RO
docsQosSfLatencyBin7Pkts	M	RO
docsQosSfLatencyBin8Pkts	M	RO
docsQosSfLatencyBin9Pkts	M	RO
docsQosSfLatencyBin10Pkts	M	RO
docsQosSfLatencyBin11Pkts	M	RO
docsQosSfLatencyBin12Pkts	M	RO
docsQosSfLatencyBin13Pkts	M	RO
docsQosSfLatencyBin14Pkts	M	RO
docsQosSfLatencyBin15Pkts	M	RO
docsQosSfLatencyBin16Pkts	M	RO
docsQosSfLatencySanctionedPkts	M	RO
docsQosSfLatencyDroppedPkts	M	RO
docsQosSfLatencyTotalEct0Pkts	M	RO
docsQosSfLatencyCeMarkedEct0Pkts	M	RO
docsQosSfLatencyTotalEct1Pkts	M	RO
docsQosSfLatencyCeMarkedEct1	M	RO
docsQosCmServiceUsStatsTable	M	N-Acc
docsQosCmServiceUsStatsEntry	M	N-Acc
docsQosCmServiceUsStatsTxSlotsImmed	M	RO
docsQosCmServiceUsStatsTxSlotsDed	M	RO
docsQosCmServiceUsStatsTxRetries	M	RO
docsQosCmServiceUsStatsTxExceededs	M	RO
docsQosCmServiceUsStatsRqRetries	M	RO
docsQosCmServiceUsStatsRqExceededs	M	RO
docsQosCmServiceUsStatsSgmts	M	RO
docsQosCmDsidTable	M	N-Acc
docsQosCmDsidEntry	M	N-Acc
docsQosCmDsidDsid	M	N-Acc
docsQosCmDsidUsage	M	RO
docsQosCmDsidNumReseqChs	M	RO

docsQosCmDsidReseqChList	M	RO
docsQosCmDsidReseqWaitTime	M	RO
docsQosCmDsidReseqWarnThrshld	M	RO
docsQosCmDsidStatusHoldOffTimerSeqOutOfRng	M	RO
docsQosCmDsidOutOfRangeDiscards	M	RO
docsQosCmDsidNextExpectedSeqNum	M	RO
docsQosCmDsidCmInterfaceMask	M	RO
docsQosCmDsidFwdCmInterfaceMask	M	RO
docsQosCmDsidStatsTable	M	N-Acc
docsQosCmDsidStatsEntry	M	N-Acc
docsQosCmDsidStatsDsid	M	N-Acc
docsQosCmDsidStatsSeqNumMissing	M	RO
docsQosCmDsidStatsSkewThreshExceeds	M	RO
docsQosCmDsidStatsOutOfRangePackets	M	RO
docsQosCmDsidStatsNumPackets	M	RO
docsQosCmDsidClientTable	M	N-Acc
docsQosCmDsidClientEntry	M	N-Acc
docsQosCmDsidClientDsid	M	N-Acc
docsQosCmDsidClientClientMacId	M	N-Acc
docsQosCmDsidClientClientMacAddr	M	RO
docsQosCmSystemCfgState		
docsQosCmSystemCfgStateAqmUsEnable	M	RO
docsQosCmSystemCfgStateDefaultUsTargetBuffer	M	RO
docsQosSfCongestionStatsTable	M	N-Acc
docsQosSfCongestionStatsEntry	M	N-Acc
docsQosSfCongestionSanctionedPkts	M	RO
docsQosSfCongestionTotalEct0Pkts	M	RO
docsQosSfCongestionTotalEct1Pkts	M	RO
docsQosSfCongestionCeMarkedEct1Pkts	M	RO
[DOCS-IF3-MIB]		
docsIf3CmStatusTable	M	N-Acc
docsIf3CmStatusEntry	M	N-Acc
docsIf3CmStatusValue	M	RO
docsIf3CmStatusCode	M	RO
docsIf3CmStatusResets	M	RO
docsIf3CmStatusLostSyncs	M	RO
docsIf3CmStatusInvalidMaps	M	RO
docsIf3CmStatusInvalidUcds	M	RO
docsIf3CmStatusInvalidRangingRsps	M	RO
docsIf3CmStatusInvalidRegRsps	M	RO
docsIf3CmStatusT1Timeouts	M	RO
docsIf3CmStatusT2Timeouts	M	RO
docsIf3CmStatusUCCsSuccesses	D	RO
docsIf3CmStatusUCCFails	D	RO
docsIf3CmStatusEnergyMgt1x1OperStatus	M	RO
docsIf3CmStatusUsTable	M	N-Acc
docsIf3CmStatusUsEntry	M	N-Acc

docsIf3CmStatusUsTxPower	M	RO
docsIf3CmStatusUsT3Timeouts	M	RO
docsIf3CmStatusUsT4Timeouts	M	RO
docsIf3CmStatusUsRangingAborteds	M	RO
docsIf3CmStatusUsModulationType	M	RO
docsIf3CmStatusUsEqData	M	RO
docsIf3CmStatusUsT3Exceededs	M	RO
docsIf3CmStatusUsIsMuted	M	RO
docsIf3CmStatusUsRangingStatus	M	RO
docsIf3CmCapabilities		
docsIf3CmCapabilitiesReq	M	RO
docsIf3CmCapabilitiesRsp	M	RO
docsIf3RxChStatusTable	M	N-Acc
docsIf3RxChStatusEntry	M	N-Acc
docsIf3RxChStatusRcId	M	N-Acc
docsIf3RxChStatusChIfIndex	M	RO
docsIf3RxChStatusPrimaryDsIndicator	M	RO
docsIf3RxChStatusRcRmConnectivityId	M	RO
docsIf3RxModuleStatusTable	M	N-Acc
docsIf3RxModuleStatusEntry	M	N-Acc
docsIf3RxModuleStatusRmId	M	N-Acc
docsIf3RxModuleStatusRmRmConnectivityId	M	RO
docsIf3RxModuleStatusFirstCenterFrequency	M	RO
docsIf3SignalQualityExtTable	M	N-Acc
docsIf3SignalQualityExtEntry	M	N-Acc
docsIf3SignalQualityExtRxMER	M	RO
docsIf3SignalQualityExtRxMerSamples	M	RO
docsIf3SignalQualityExtFbeNormalizationCoefficient	M	RO
docsIf3UsChExtTable	M	N-Acc
docsIf3UsChExtEntry	M	N-Acc
docsIf3UsChExtSacCodeHoppingSelectionMode	M	RO
docsIf3UsChExtScdmaSelectionStringActiveCodes	O	RO
docsIf3CmDpvStatsTable	M	N-Acc
docsIf3CmDpvStatsEntry	M	N-Acc
docsIf3CmDpvStatsGrpId	M	N-Acc
docsIf3CmDpvStatsLastMeasLatency	M	RO
docsIf3CmDpvStatsLastMeasTime	M	RO
docsIf3CmDpvStatsMinLatency	M	RO
docsIf3CmDpvStatsMaxLatency	M	RO
docsIf3CmDpvStatsAvgLatency	M	RO
docsIf3CmDpvStatsNumMeas	M	RO
docsIf3CmDpvStatsLastClearTime	M	RO
docsIf3CmEventCtrlTable	M	N-Acc
docsIf3CmEventCtrlEntry	M	N-Acc
docsIf3CmEventCtrlEventId	M	N-Acc
docsIf3CmEventCtrlStatus	M	RC
docsIf3CmMdCfgTable	M	N-Acc

docsIf3CmMdCfgEntry	M	N-Acc
docsIf3CmMdCfgIpProvMode	M	RW
docsIf3CmMdCfgIpProvModeResetOnChange	M	RW
docsIf3CmMdCfgIpProvModeResetOnChangeHoldOffTimer	M	RW
docsIf3CmMdCfgIpProvModeStorageType	M	RW
docsIf3CmEnergyMgtCfg		
docsIf3CmEnergyMgtCfgFeatureEnabled	M	RO
docsIf3CmEnergyMgtCfgCyclePeriod	M	RO
docsIf3CmEnergyMgt1x1CfgTable	M	N-Acc
docsIf3CmEnergyMgt1x1CfgEntry	M	N-Acc
docsIf3CmEnergyMgt1x1CfgDirection	M	N-Acc
docsIf3CmEnergyMgt1x1CfgEntryBitrateThrshld	M	RW
docsIf3CmEnergyMgt1x1CfgEntryTimeThrshld	M	RW
docsIf3CmEnergyMgt1x1CfgExitBitrateThrshld	M	RW
docsIf3CmEnergyMgt1x1CfgExitTimeThrshld	M	RW
docsIf3CmSpectrumAnalysisCtrlCmd		
docsIf3CmSpectrumAnalysisCtrlCmdEnable	M	RW
docsIf3CmSpectrumAnalysisCtrlCmdInactivityTimeout	M	RW
docsIf3CmSpectrumAnalysisCtrlCmdFirstSegmentCenterFrequency	M	RO/RW
docsIf3CmSpectrumAnalysisCtrlCmdLastSegmentCenterFrequency	M	RO/RW
docsIf3CmSpectrumAnalysisCtrlCmdSegmentFrequencySpan	M	RO/RW
docsIf3CmSpectrumAnalysisCtrlCmdNumBinsPerSegment	M	RO/RW
docsIf3CmSpectrumAnalysisCtrlCmdEquivalentNoiseBandwidth	M	RO/RW
docsIf3CmSpectrumAnalysisCtrlCmdWindowFunction	M	RO/RW
docsIf3CmSpectrumAnalysisCtrlCmdNumberOfAverages	M	RO/RW
docsIf3CmSpectrumAnalysisMeasTable	M	N-Acc
docsIf3CmSpectrumAnalysisMeasEntry	M	N-Acc
docsIf3CmSpectrumAnalysisMeasFrequency	M	N-Acc
docsIf3CmSpectrumAnalysisMeasAmplitudeData	M	RO
docsIf3CmSpectrumAnalysisMeasTotalSegmentPower	M	RO
docsIf3CmEm1x1StatsTable	M	N-Acc
docsIf3CmEm1x1StatsEntry	M	N-Acc
docsIf3CmEm1x1StatsNumberTimesCrossedBelowUsEntryThrshlds	M	RO
docsIf3CmEm1x1StatsNumberTimesCrossedBelowDsEntryThrshlds	M	RO
docsIf3CmEm1x1StatsTotalDuration	M	RO
docsIf3CmEm1x1StatsTotalDurationBelowUsThrshlds	M	RO
docsIf3CmEm1x1StatsTotalDurationBelowDsThrshlds	M	RO
docsIf3CmEm1x1StatsTotalDurationBelowUsDsThrshlds	M	RO
Notifications		
docsIf3CmEventNotif	M	Notif
[DOCS-IF31-MIB]		
docsIf31DocsisBaseCapability	M	RO
docsIf31RxChStatusTable	M	N-Acc
docsIf31RxChStatusEntry	M	N-Acc
docsIf31RxChStatusPrimaryDsIndicator	M	RO
docsIf31RxChStatusOfdmProfiles	M	RO
docsIf31CmDsOfdmChanTable	M	N-Acc

docsIf31CmDsOfdmChanEntry	M	N-Acc
docsIf31CmDsOfdmChanChannelId	M	RO
docsIf31CmDsOfdmChanChanIndicator	M	RO
docsIf31CmDsOfdmChanSubcarrierZeroFreq	M	RO
docsIf31CmDsOfdmChanFirstActiveSubcarrierNum	M	RO
docsIf31CmDsOfdmChanLastActiveSubcarrierNum	M	RO
docsIf31CmDsOfdmChanNumActiveSubcarriers	M	RO
docsIf31CmDsOfdmChanSubcarrierSpacing	M	RO
docsIf31CmDsOfdmChanCyclicPrefix	M	RO
docsIf31CmDsOfdmChanRollOffPeriod	M	RO
docsIf31CmDsOfdmChanPlcFreq	M	RO
docsIf31CmDsOfdmChanNumPilots	M	RO
docsIf31CmDsOfdmChanTimeInterleaverDepth	M	RO
docsIf31CmDsOfdmChanPlcTotalCodewords	M	RO
docsIf31CmDsOfdmChanPlcUnreliableCodewords	M	RO
docsIf31CmDsOfdmChanNcpTotalFields	M	RO
docsIf31CmDsOfdmChanNcpFieldCrcFailures	M	RO
docsIf31CmDsOfdmProfileStatsTable	M	N-Acc
docsIf31CmDsOfdmProfileStatsEntry	M	N-Acc
docsIf31CmDsOfdmProfileStatsProfileId	M	N-Acc
docsIf31CmDsOfdmProfileStatsConfigChangeCt	M	RO
docsIf31CmDsOfdmProfileStatsTotalCodewords	M	RO
docsIf31CmDsOfdmProfileStatsCorrectedCodewords	M	RO
docsIf31CmDsOfdmProfileStatsUncorrectableCodewords	M	RO
docsIf31CmDsOfdmProfileStatsInOctets	M	RO
docsIf31CmDsOfdmProfileStatsInUnicastOctets	M	RO
docsIf31CmDsOfdmProfileStatsInMulticastOctets	M	RO
docsIf31CmDsOfdmProfileStatsInFrames	M	RO
docsIf31CmDsOfdmProfileStatsInUnicastFrames	M	RO
docsIf31CmDsOfdmProfileStatsInMulticastFrames	M	RO
docsIf31CmDsOfdmProfileStatsInFrameCrcFailures	M	RO
docsIf31CmDsOfdmProfileStatsCtrDiscontinuityTime	M	RO
docsIf31CmDsOfdmChannelPowerTable	M	N-Acc
docsIf31CmDsOfdmChannelPowerEntry	M	N-Acc
docsIf31CmDsOfdmChannelBandIndex	M	N-Acc
docsIf31CmDsOfdmChannelPowerCenterFrequency	M	RO
docsIf31CmDsOfdmChannelPowerRxPower	M	RO
docsIf31CmStatusOfdmaUsTable	M	N-Acc
docsIf31CmStatusOfdmaUsEntry	M	N-Acc
docsIf31CmStatusOfdmaUsT3Timeouts	M	RO
docsIf31CmStatusOfdmaUsT4Timeouts	M	RO
docsIf31CmStatusOfdmaUsRangingAborteds	M	RO
docsIf31CmStatusOfdmaUsT3Exceededs	M	RO
docsIf31CmStatusOfdmaUsIsMuted	M	RO
docsIf31CmStatusOfdmaUsRangingStatus	M	RO
docsIf31CmUsOfdmaChanTable	M	N-Acc
docsIf31CmUsOfdmaChanEntry	M	N-Acc

docsIf31CmStatusOfdmaChanConfigChangeCt	M	RO
docsIf31CmUsOfdmaChanSubcarrierZeroFreq	M	RO
docsIf31CmUsOfdmaChanFirstActiveSubcarrierNum	M	RO
docsIf31CmUsOfdmaChanLastActiveSubcarrierNum	M	RO
docsIf31CmUsOfdmaChanNumActiveSubcarriers	M	RO
docsIf31CmUsOfdmaChanSubcarrierSpacing	M	RO
docsIf31CmUsOfdmaChanCyclicPrefix	M	RO
docsIf31CmUsOfdmaChanRollOffPeriod	M	RO
docsIf31CmUsOfdmaChanNumSymbolsPerFrame	M	RO
docsIf31CmUsOfdmaChanTxPower	M	RO
docsIf31CmUsOfdmaChanPreEqEnabled	M	RO
docsIf31CmUsOfdmaChanChannelId	M	RO
docsIf31CmUsScQamChanTable	M	N-Acc
docsIf31CmUsScQamChanEntry	M	N-Acc
docsIf31CmUsScQamChanTxPsd	M	RO
docsIf31CmUsOfdmaProfileStatsTable	M	N-Acc
docsIf31CmUsOfdmaProfileStatsEntry	M	N-Acc
docsIf31CmUsOfdmaProfileStatsIuc	M	N-Acc
docsIf31CmUsOfdmaProfileStatsOutOctets	M	RO
docsIf31CmUsOfdmaProfileStatsCtrDiscontinuityTime	M	RO
docsIf31CmUsOfdmaMinislotCfgStateTable	M	N-Acc
docsIf31CmUsOfdmaMinislotCfgStateEntry	M	N-Acc
docsIf31CmUsOfdmaMinislotCfgStateStartMinislotNum	M	N-Acc
docsIf31CmUsOfdmaMinislotCfgStateFirstSubcarrierId	M	RO
docsIf31CmUsOfdmaMinislotCfgStateNumConsecutiveMinislots	M	RO
docsIf31CmUsOfdmaMinislotCfgStateMinislotPilotPattern	M	RO
docsIf31CmUsOfdmaMinislotCfgStateDataSymbolModulation	M	RO
docsIf31CmEmDlsStatsTable	M	N-Acc
docsIf31CmEmDlsStatsEntry	M	N-Acc
docsIf31CmEmDlsStatsNumberTimesCrossedBelowUsEntryThrshlds	M	RO
docsIf31CmEmDlsStatsNumberTimesCrossedBelowDsEntryThrshlds	M	RO
docsIf31CmEmDlsStatsTotalDuration	M	RO
docsIf31CmEmDlsStatsTotalDurationBelowUsThrshlds	M	RO
docsIf31CmEmDlsStatsTotalDurationBelowDsThrshlds	M	RO
docsIf31CmEmDlsStatsTotalDurationBelowUsDsThrshlds	M	RO
docsIf31CmEmDlsStatsNumSleepLatencyTriggers	M	RO
docsIf31CmEmDlsStatsNumSleepByteCtTriggers	M	RO
docsIf31CmEmDlsStatusTable	M	N-Acc
docsIf31CmEmDlsStatusEntry	M	N-Acc
docsIf31CmEmDlsStatusAssignedEmIds	M	N-Acc
docsIf31CmEmDlsStatusReceiveTimer	M	RO
docsIf31CmEmDlsStatusMaxSleepLatency	M	RO
docsIf31CmEmDlsStatusMaxSleepBytes	M	RO
docsIf31CmSystemCfgState	M	N-Acc
docsIf31CmSystemCfgStateDiplexerCapability	M	RO
docsIf31CmSystemCfgStateDiplexerCfgBandEdge	M	RO
docsIf31CmSystemCfgStateDiplexerDsLowerCapability	M	RO

docsIf31CmSystemCfgStateDiplexerCfgDsLowerBandEdge	M	RO
docsIf31CmSystemCfgStateDiplexerDsUpperCapability	M	RO
docsIf31CmSystemCfgStateDiplexerCfgDsUpperBandEdge	M	RO
docsIf31CmFddSystemCfgStateAdvDsLowerBandEdge	M	RO
docsIf31CmFddSystemCfgStateAdvDsUpperBandEdge	M	RO
docsIf31CmFddSystemCfgStateAdvUsUpperBandEdge	M	RO
docsIf31CmStatusTable	M	N-Acc
docsIf31CmStatusEntry	M	N-Acc
docsIf31CmStatusEmDlsOperStatus	M	RO
docsIf31CmEmDlsCfgTable	M	N-Acc
docsIf31CmEmDlsCfgEntry	M	N-Acc
docsIf31CmEmDlsCfgDirection	M	N-Acc
docsIf31CmEmDlsCfgEntryBitrateThrshld	M	RW
docsIf31CmEmDlsCfgEntryTimeThrshld	M	RW
docsIf31CmEmDlsCfgExitBitrateThrshld	M	RW
docsIf31CmEmDlsCfgExitTimeThrshld	M	RW
[DOCS-PNM-MIB]		
docsPnmBulkCtl Group		
docsPnmBulkDestIpAddrType	M	RW
docsPnmBulkDestIpAddr	M	RW
docsPnmBulkDestPath	M	RW
docsPnmBulkUploadControl	M	RW
docsPnmBulkFileTable	M	N-Acc
docsPnmBulkFileEntry	M	N-Acc
docsPnmBulkFileIndex	M	N-Acc
docsPnmBulkFileName	M	RO
docsPnmBulkFileControl	M	RW
docsPnmBulkFileUploadStatus	M	RO
docsPnmCmControlObjects Group		
docsPnmCmCtlTest	M	RO
docsPnmCmCtlTestDuration	M	RO
docsPnmCmCtlStatus	M	RO
docsPnmCmDsOfdmSymCapTable	M	N-Acc
docsPnmCmDsOfdmSymCapEntry	M	N-Acc
docsPnmCmDsOfdmSymTrigEnable	M	RW
docsPnmCmDsOfdmSymTrigEnableTimeout	M	RW
docsPnmCmDsOfdmSymTrigGroupId	M	RW
docsPnmCmDsOfdmSymRxWindowing	M	RO
docsPnmCmDsOfdmSymTransactionId	M	RO
docsPnmCmDsOfdmSymSampleRate	M	RO
docsPnmCmDsOfdmSymFftLength	M	RO
docsPnmCmDsOfdmSymMeasStatus	M	RO
docsPnmCmDsOfdmSymCaptFileName	M	RW
docsPnmCmOfdmChanEstCoefTable	M	N-Acc
docsPnmCmOfdmChanEstCoefEntry	M	N-Acc
docsPnmCmOfdmChEstCoefTrigEnable	M	RW
docsPnmCmOfdmChEstCoefAmpRipplePkToPk	M	RO

docsPnmCmOfdmChEstCoefAmpRippleRms	M	RO
docsPnmCmOfdmChEstCoefAmpSlope	M	RO
docsPnmCmOfdmChEstCoefGrpDelayRipplePkToPk	M	RO
docsPnmCmOfdmChEstCoefGrpDelayRippleRms	M	RO
docsPnmCmOfdmChEstCoefMeasStatus	M	RO
docsPnmCmOfdmChEstCoefFileName	M	RW
docsPnmCmOfdmChEstCoefAmpMean	M	RO
docsPnmCmOfdmChEstCoefGrpDelaySlope	M	RO
docsPnmCmOfdmChEstCoefGrpDelayMean	M	RO
docsPnmCmDsConstDispMeasTable	M	N-Acc
docsPnmCmDsConstDispMeasEntry	M	N-Acc
docsPnmCmDsConstDispTrigEnable	M	RW
docsPnmCmDsConstDispModOrderOffset	M	RW
docsPnmCmDsConstDispNumSampleSymb	M	RW
docsPnmCmDsConstDispSelModOrder	M	RO
docsPnmCmDsConstDispMeasStatus	M	RO
docsPnmCmDsConstDispFileName	M	RW
docsPnmCmDsOfdmRxMerTable	M	N-Acc
docsPnmCmDsOfdmRxMerEntry	M	N-Acc
docsPnmCmDsOfdmRxMerFileEnable	M	RW
docsPnmCmDsOfdmRxMerPercentile	M	RW
docsPnmCmDsOfdmRxMerMean	M	RO
docsPnmCmDsOfdmRxMerStdDev	M	RO
docsPnmCmDsOfdmRxMerThrVal	M	RO
docsPnmCmDsOfdmRxMerThrHighestFreq	M	RO
docsPnmCmDsOfdmRxMerMeasStatus	M	RO
docsPnmCmDsOfdmRxMerFileName	M	RW
docsPnmCmDsOfdmMerMarTable	M	N-Acc
docsPnmCmDsOfdmMerMarEntry	M	N-Acc
docsPnmCmDsOfdmMerMarProfileId	M	RW
docsPnmCmDsOfdmMerMarThrshldOffset	M	RW
docsPnmCmDsOfdmMerMarMeasEnable	M	RW
docsPnmCmDsOfdmMerMarNumSymPerSubCarToAvg	M	RW
docsPnmCmDsOfdmMerMarReqAvgMer	M	RW
docsPnmCmDsOfdmMerMarNumSubCarBelowThrshld	M	RO
docsPnmCmDsOfdmMerMarMeasuredAvgMer	M	RO
docsPnmCmDsOfdmMerMarAvgMerMargin	M	RO
docsPnmCmDsOfdmMerMarMeasStatus	M	RO
docsPnmCmDsOfdmFecTable	M	N-Acc
docsPnmCmDsOfdmFecEntry	M	N-Acc
docsPnmCmDsOfdmFecSumType	M	RW
docsPnmCmDsOfdmFecFileEnable	M	RW
docsPnmCmDsOfdmFecMeasStatus	M	RO
docsPnmCmDsOfdmFecFileName	M	RW
docsPnmCmDsOfdmReqMERObjects		
docsPnmCmDsOfdmReqMerQam16	M	RW
docsPnmCmDsOfdmReqMerQam64	M	RW

docsPnmCmDsOfdmReqMerQam128	M	RW
docsPnmCmDsOfdmReqMerQam256	M	RW
docsPnmCmDsOfdmReqMerQam512	M	RW
docsPnmCmDsOfdmReqMerQam1024	M	RW
docsPnmCmDsOfdmReqMerQam2048	M	RW
docsPnmCmDsOfdmReqMerQam4096	M	RW
docsPnmCmDsOfdmReqMerQam8192	M	RW
docsPnmCmDsOfdmReqMerQam16384	M	RW
docsPnmCmDsHistTable	M	N-Acc
docsPnmCmDsHistEntry	M	N-Acc
docsPnmCmDsHistEnable	M	RW
docsPnmCmDsHistTimeOut	M	RW
docsPnmCmDsHistMeasStatus	M	RO
docsPnmCmDsHistFileName	M	RW
docsPnmCmUsPreEqTable	M	N-Acc
docsPnmCmUsPreEqEntry	M	N-Acc
docsPnmCmUsPreEqFileEnable	M	RW
docsPnmCmUsPreEqAmpRipplePkToPk	M	RO
docsPnmCmUsPreEqAmpRippleRms	M	RO
docsPnmCmUsPreEqAmpSlope	M	RO
docsPnmCmUsPreEqGrpDelayRipplePkToPk	M	RO
docsPnmCmUsPreEqGrpDelayRippleRms	M	RO
docsPnmCmUsPreEqPreEqCoAdjStatus	M	RO
docsPnmCmUsPreEqMeasStatus	M	RO
docsPnmCmUsPreEqLastUpdateFileName	M	RW
docsPnmCmUsPreEqFileName	M	RW
docsPnmCmUsPreEqAmpMean	M	RO
docsPnmCmUsPreEqGrpDelaySlope	M	RO
docsPnmCmUsPreEqGrpDelayMean	M	RO
docsPnmCmDsOfdmModProfTable	M	N-Acc
docsPnmCmDsOfdmModProfEntry	M	N-Acc
docsPnmCmDsOfdmModProfFileEnable	M	RW
docsPnmCmDsOfdmModProfMeasStatus	M	RO
docsPnmCmDsOfdmModProfFileName	M	RW
docsCmLatencyRptCfgSnapshotDuration	M	RW
docsCmLatencyRptCfgNumSnapshots	M	RW
docsCmLatencyRptCfgNumFiles	M	RW
docsCmLatencyRptCfgMeasStatus	M	RO
docsCmLatencyRptCfgFileName	M	RW
docsIf3CmSpectrumAnalysis		
docsIf3CmSpectrumAnalysisCtrlCmdFileEnable	M	RW
docsIf3CmSpectrumAnalysisCtrlCmdMeasStatus	M	RO
docsIf3CmSpectrumAnalysisCtrlCmdFileName	M	RW
[DOCS-BPI2EXT-MIB]		
docsBpi2Ext31CmDeviceCertTable	M	N-Acc
docsBpi2Ext31CmDeviceCertEntry	M	N-Acc
docsBpi2Ext31CmDeviceCmCert	M	RW

docsBpi2Ext31CmDeviceManufCert	M	RO
docsBpi2Ext31CodeDownloadControl		
docsBpi2Ext31CodeUpdateCvcChain	M	RW
docsBpi2Ext31CodeMfgOrgName	M	RO
docsBpi2Ext31CodeMfgCodeAccessStart	M	RO
docsBpi2Ext31CodeMfgCvcAccessStart	M	RO
docsBpi2Ext31CodeCoSignerOrgName	M	RO
docsBpi2Ext31CodeCoSignerCodeAccessStart	M	RO
docsBpi2Ext31CodeCoSignerCvcAccessStart	M	RO
[DOCS-SEC-MIB]		
docsSecCmSshServer		
docsSecCmSshServerEnabledInterfaces	M	RW
docsSecCmSshServerStatus	M	RO
docsSecCmSshServerPublicKey	M	RO
docsSecCmSshServerNewConnectionTimeout	M	RW
docsSecCmSshServerInactivityTimeout	M	RW
docsSecCmSshServerSshSourceAddrRestrictionType	O	RW
docsSecCmSshServerSshSourceAddrRestriction	O	RW
docsSecCmSshServerSshSourcePrefixRestriction	O	RW
docsSecCmCdsFileServer		
docsSecCmCdsFileServerIpAddrType	M	RW
docsSecCmCdsFileServerIpAddr	M	RW
docsSecCmCdsFileServerSshCmCdsDownloadUrl	M	RW
docsSecCmCdsFileServerRevocationStatusAction	M	RW
docsSecCmPasswordCredentialTable	M	N-Acc
docsSecCmPasswordCredentialEntry	M	N-Acc
docsSecCmPasswordCredentialIndex	M	N-Acc
docsSecCmPasswordCredentialUserId	M	RC
docsSecCmPasswordCredentialPassword	M	RC
docsSecCmPasswordCredentialMacAddr	O	RC
docsSecCmPasswordCredentialRowStatus	M	RC
docsSecCmPublicKeyCredentialTable	M	N-Acc
docsSecCmPublicKeyCredentialEntry	M	N-Acc
docsSecCmPublicKeyCredentialIndex	M	N-Acc
docsSecCmPublicKeyCredentialSshPublicKey	M	RC
docsSecCmPublicKeyCredentialMacAddr	O	RC
docsSecCmPublicKeyCredentialRowStatus	M	RC
docsSecCmSccaServerCfg		
docsSecCmSccaServerCfgIpAddrType	M	RW
docsSecCmSccaServerCfgIpAddr	M	RW
docsSecCmSccaServerCfgRestApiUrl	M	RW
docsSecCmSccaServerCfgRevocationStatusAction	M	RW

A.2 RFC 2863 ifTable/ifXTable MIB-Object Details

Refer to [RFC 2863] for MIB object descriptions. Table 71 includes DOCSIS 4.0 specific object information.

The following tables detail the specific ifTable and ifXTable MIB objects and values that are expected for the interfaces on the CM.

Section 7.1.3.8.5 has defined the requirements for the [RFC 2863] ifTable and ifXTable MIB objects. This section applies these general requirements to the CM interfaces. Table 74 defines the specific requirements for the CM ethernet, USB and other interfaces. Table 75 defines the specific requirements for the CM upstream, downstream and MAC interfaces. Table 74 and Table 75 exclude the Counter32 and Counter64 MIB objects as these counter objects are defined in Table 76 and Table 77.

In order to simplify and compile all the requirements for the Counter32 and Counter64 MIB objects in a single location, the specific SNMP Access requirements and MIB implementation details that are normally detailed in Annex A.1 are reflected in Table 76 and Table 77. The nomenclature for the MIB implementation details can be found in Table 71 and the SNMP Access Requirements are detailed in Table 72. Please refer to these tables for the values found for each of the interfaces in Table 76 and Table 77.

In addition to the requirements for Ethernet and USB detailed in Table 74 - [RFC 2863] ifTable/ifXTable MIB-Object Details for Ethernet and USB Interfaces below, note that the various packet and octet counters from the ifTable and ifXTable MAY exclude LAN-LAN traffic which is not bridged upstream or downstream. From the ifTable, these counters include the following: ifInOctets, ifInUcastPkts, ifOutOctets, and ifOutUcastPkts. From the ifXTable, included counters are ifInMulticastPkts, ifInBroadcastPkts, ifOutMulticastPkts, ifOutBroadcastPkts, ifHCInOctets, ifHCInUcastPkts, ifHCInMulticastPkts, ifHCInBroadcastPkts, ifHCOutOctets, ifHCOutUcastPkts, ifHCOutMulticastPkts, and ifHCOutBroadcastPkts.

Table 74 - [RFC 2863] ifTable/ifXTable MIB-Object Details for Ethernet and USB Interfaces

MIB Objects	CM-Ethernet	CM USB CDC Ethernet	CM-CPE OtherType
IfTable			
ifIndex	1 or [4+(n)]	1 or [4+(n)]	1 or [4+(n)]
ifDescr		See Section 7.1.3.8.7.1	
ifType	6	160	(IANA num)
ifMtu	1500	1500	Media dependent
ifSpeed	10,000,000, 100,000,000, ...	12,000,000, 480,000,000	speed
ifPhysAddress	MAC Address of this interface	MAC Address of this interface	Media dependent
ifAdminStatus	up(1), down(2), testing(3)	up(1), down(2), testing(3)	up(1), down(2), testing(3)
ifOperStatus	up(1), down(2), testing(3), dormant(5), notPresent(6)	See Section 7.1.3.8.2.2	up(1), down(2), testing(3), dormant(5), notPresent(6)
ifLastChange			
ifXTable			
ifName			
ifLinkUpDownTrapEnable **Note:** See Section 7.1.3.8.4 for details			
ifHighSpeed	10, 100, ...	12, 480	speed
ifPromiscuousMode	true, false	true, false	true, false
ifConnectorPresent			
ifAlias			
ifCounterDiscontinuityTime			

Note: Refer to Table 77 for Counter32 and Counter64 MIB object details.

Table 75 - [RFC 2863] ifTable/ifXTable MIB-Object Details for MAC and RF Interfaces

MIB Objects	CM-MAC	CM-Downstream	CM-Upstream
IfTable			
ifIndex	2	3	4
ifDescr			
ifType	127	128 (SC-QAM) 277 (OFDM)	129 (SC-QAM) 278 (OFDMA)
ifMtu (For RF Upstream/Downstream; the value includes the length of the MAC header.) **Note:** The ifMtu MIB reports a value based on the value returned in the Extended Packet Length Support modem capability [MULPIv4.0]. In the case of the CM-Downstream and CM-Upstream, the reported value of ifMTU additionally accounts for the length of the DOCSIS MAC header.	1522 (if Extended Packet Length Support modem capability is disabled) Value of Extended Packet Length Support modem capability (TLV 5.48) (if Extended Packet Length Support modem capability is enabled)	1764 (if Extended Packet Length Support modem capability is disabled) Value of Extended Packet Length Support modem capability (TLV 5.48) + 30 (if Extended Packet Length Support modem capability is enabled)	1764 (if Extended Packet Length Support modem capability is disabled Value of Extended Packet Length Support modem capability (TLV 5.48) + 30 (if Extended Packet Length Support modem capability is enabled)
ifSpeed	0	Refer to Section 7.1.3.8.6	Refer to Section 7.1.3.8.6
ifPhysAddress:	MAC Address of this interface	Empty-String	Empty-String
ifAdminStatus: Refer to Section 7.1.3.8.3	up(1), down(2), testing(3)	up(1), down(2), testing(3)	up(1), down(2), testing(3)
ifOperStatus: Refer to Section 7.1.3.8.2	up(1), down(2), testing(3), dormant(5), notPresent(6)	up(1), down(2), testing(3), dormant(5), notPresent(6)	up(1), down(2), testing(3), dormant(5), notPresent(6)
ifLastChange:			
ifXTable			
ifName			
ifLinkUpDownTrapEnable See Section 7.1.3.8.5.			
ifHighSpeed	0	Refer to Section 7.1.3.8.6	Refer to Section 7.1.3.8.6
ifPromiscuousMode	true	true	false
ifConnectorPresent			
ifAlias			
ifCounterDiscontinuityTime			

Note: Refer to Table 77 for Counter32 and Counter64 MIB object details.

Table 76 - [RFC 2863] ifTable/ifXTable Counter32 and Counter64 MIB-Object Details for Ethernet and USB Interfaces

MIB Counter Objects	ACCESS	CM-Ethernet	CM-USB	CM-CPE OtherType
ifTable				
ifInOctets	RO	M	M	M
ifInUcastPkts	RO	M	M	M
ifInDiscards	RO	M	M	M

MIB Counter Objects	ACCESS	CM-Ethernet	CM-USB	CM-CPE OtherType
ifInErrors	RO	O	O	M
ifInUnknownProtos	RO	M	M	M
ifOutOctets	RO	M	M	M
ifOutUcastPkts	RO	M	M	M
ifOutDiscards	RO	M	M	M
ifOutErrors	RO	M	M	M
ifXTable				
ifInMulticastPkts	RO	M	M	M
ifInBroadcastPkts	RO	M	M	M
ifOutMulticastPkts	RO	M	M	M
ifOutBroadcastPkts	RO	M	M	M
IfHCInOctets	RO	O	O	O
ifHCInUcastPkts	RO	O	O	O
ifHCInMulticastPkts	RO	O	O	O
ifHCInBroadcastPkts	RO	O	O	O
ifHCOutOctets	RO	O	O	O
ifHCOutUcastPkts	RO	O	O	O
ifHCOutMulticastPkts	RO	O	O	O
ifHCOutBroadcastPkts	RO	O	O	O

In Table 77, the packet and octet counters are implemented based on the requirements in Section 7 of this specification. In this table, the value NA means that the particular counter is not applicable to this interface. Objects labeled as NA or O in Table 77 can be optionally implemented and if implemented, the object will return 0 when read.

Table 77 - [RFC 2863] ifTable/ifXTable Counter32 and Counter64 MIB-Object Details for MAC and RF Interfaces

MIB Counter Objects	Access	CM-MAC	CM-Downstream	CM-Upstream
ifTable				
ifInOctets	RO	M	M	NA
ifInUcastPkts	RO	M	O	NA
ifInDiscards	RO	M	O	NA
ifInErrors	RO	M	O	NA
ifInUnknownProtos	RO	M	O	NA
ifOutOctets	RO	M	NA	M
ifOutUcastPkts	RO	M	NA	O
ifOutDiscards	RO	M	NA	O
ifOutErrors	RO	M	NA	O
ifXTable				
ifInMulticastPkts	RO	M	O	NA
ifInBroadcastPkts	RO	M	O	NA
ifOutMulticastPkts	RO	M	NA	O
ifOutBroadcastPkts	RO	M	NA	O
IfHCInOctets	RO	M	M	NA
ifHCInUcastPkts	RO	O	O	NA
ifHCInMulticastPkts	RO	O	O	NA

MIB Counter Objects	Access	CM-MAC	CM-Downstream	CM-Upstream
ifHCInBroadcastPkts	RO	O	O	NA
ifHCOutOctets	RO	M	NA	M
ifHCOutUcastPkts	RO	O	NA	O
ifHCOutMulticastPkts	RO	O	NA	O
ifHCOutBroadcastPkts	RW	O	NA	O

Annex B IP Protocol and LLC Filtering and Classification (Normative)

DOCSIS 3.0 CMs supported two packet filtering/classification methods consisting of the legacy IP filtering mechanism specified in [RFC 4639] and Upstream Drop Classifiers (UDCs). A DOCSIS 4.0 CM is only required to support UDCs and is not required to support legacy [RFC 4639] IP and LLC filtering mechanisms.

UDCs are modeled on the existing QoS Classifiers that were introduced in DOCSIS 1.1. UDCs apply only to the CM, the RF interface and only in the upstream direction of flow. The use of UDCs facilitates delegation of upstream protocol filtering at the CM through parameters in the configuration file that can be controlled by the CMTS. Any packet classified by the Upstream Drop Classifier rule is discarded, conceptually similarly to directing an IP route to "null 0" or output to /dev/null in a UNIX system.

As with legacy IP filters, UDC rules may be configured through the CM configuration file statically, assigned dynamically from the CMTS through a Group ID reference in the CM configuration file, dynamically added, changed or deleted after registration through a DSC (Dynamic Service Change) MAC management message from the CMTS, or both the static and dynamic configuration methods may be used together. The CMTS alone provides the downstream protocol filtering/classification and can further reinforce the upstream classification policy through Subscriber Management traffic filtering functionality if desired.

Among the specific requirements for classification at the CM, the CM is required to perform protocol classification from the host CPE(s) to the RF interface when UDCs are enabled. All IP and LLC packets will be forwarded from the CMCI interface to the RFI upstream interface based on rules outlined in the Upstream Drop Classifiers section of [MULPIv4.0], unless they are specifically required to be discarded according to applied protocol classification rules.

B.1 Filtering Mechanisms

The legacy DOCSIS filters are subdivided into two (2) filtering layers (LLC and IP) at the CM. The two legacy classification/filtering tables at the CM are the docsDevFilterIpTable and the docsDevFilterLlcTable. Upstream Drop Classifiers cover both the LLC and IP packet criterion, matching the functionality of the legacy filtering mechanisms.

B.1.1 LLC Filters

A DOCSIS 4.0 CM is not required to support RFC 4639 LLC Filters.

B.1.2 Special filters

Special filters include IP spoofing filters, inter-eSAFE and eSAFE to CPE communications and SNMP access filters such as SNMPv1/v2c NmAccess mode (see Section 8.5.2.2) and SNMP CPE Access Control (see Section 8.5.2.9).

B.1.2.1 IP Spoofing Filters

A DOCSIS 4.0 CM is not required to support [RFC 4639] IP Spoofing Filters.

B.1.2.2 Additional requirement on dot1dTpFdbTable (RFC 4188)

CM CPE MAC addresses learned via the CM configuration file MUST set the dot1dTpFdbStatus to "mgmt". It is assumed that the number of "mgmt"-configured CM CPE MAC addresses is less than, or equal to, the TLV type-18 value (Maximum Number of CPE).

B.1.2.3 SNMP Access Filter

When the CM is operating in SNMPv1/v2c NmAccess mode, the CM MUST apply the SNMP access filters to SNMP packets entering from any interface and destined for the CM. The CM MUST apply SNMP access filters after IP spoofing filters for the packets entering the CM from the CMCI interface. Since SNMP access filter function is controlled by docsDevNmAccessTable, SNMP access filter is available and applies only when the CM is in SNMP v1/v2c NmAccess mode.

When the CM is operating in SNMP Coexistence mode, SNMP access MUST be controlled and specified by the MIB objects in [RFC 3411] through [RFC 3415], and [RFC 3584].

CMs may have multiple interfaces. If SNMP access filters are applied to CM IfIndex 1, the CM MUST apply the same filters to the "Additional CPE interfaces" (see Section 7.1.3.8.1).

B.1.2.3.1 docsDevNmAccessIP and docsDevNmAccessIpMask

A CM that implements docsDevNmAccessTable MUST apply the following rules in order to determine whether to permit SNMP access from a given source IP address (SrcIpAddr):

1. If (docsDevNmAccessIp == "255.255.255.255"), the CM MUST permit the access from any SrcIpAddr.
2. If ((docsDevNmAccessIp AND docsDevNmAccessIpMask) == (SrcIpAddr AND docsDevNmAccessIpMask)), the CM MUST permit the access from SrcIpAddr.
3. If (docsDevNmAccessIp == "0.0.0.0" AND docsDevNmAccessIpMask != '255.255.255.255'), the CM MUST permit access from any SrcIpAddr.
4. If neither #1, #2, or #3 is applied, the CM MUST NOT permit the access from SrcIpAddr.

The CM's default value of the docsDevNmAccessIpMask MUST be set to "0.0.0.0".

The following table contains sample MIB values and the access granted by those values.

Table 78 - Sample docsDevNmAccessIp Values

docsDevNmAccessIp	docsDevNmAccessIpMask	Access
255.255.255.255	Any IP Address Mask	Any NMS
Any IP Address	0.0.0.0	Any NMS
Any IP Address except 255.255.255.255	255.255.255.255	Single NMS
0.0.0.0	255.255.255.255	No NMS (disables all access)
0.0.0.0	Any IP Address Mask except 255.255.255.255	Any NMS

If the CMTS implements docsDevNmAccessTable, the same rules as stated above for the CM are followed.

B.1.3 IP Protocol Filtering

A DOCSIS 4.0 CM is not required to support RFC 4639 IP Filters.

B.1.4 Protocol Classification Through Upstream Drop Classifiers

The Upstream Drop Classifier (UDC) is a structural convention re-using the definition of upstream classifiers from [MULPIv4.0]. A unique top-level TLV (Upstream Drop Packet Classification Encoding, TLV 60) defines UDCs and distinguishes this type of classifier from the QoS classifier type (Upstream Packet Classification Encoding, TLV 22). UDCs are used to discard a packet matched to the classifier rule criteria. See the Upstream Drop Packet Classification Encoding section in the Common Radio Frequency Interface Encodings Annex of [MULPIv4.0] for more details.

The CM is required to support a minimum of 64 UDC rules.

The Upstream Drop Classifier configuration structure is strictly designed to discard packets before they reach the output queue of the RFI interface and does not require attributes such as QoS. Upstream Drop Classifiers have a many-to-one relationship between UDC rules and the packet discard function. UDCs operate only within the local context of the CM. Any packet matched by a classifier rule is immediately discarded.

The CM will ignore UDC parameters which are incompatible with the packet discard function when they are configured in the CM configuration file.

B.1.4.1 IP Classification Rule Order Priority

QoS rule priority generally supersedes drop rules, though this is a configuration decision and not dictated in these specifications. For example, during a viral outbreak or DoS attack, it may be preferable to apply drop rules with

higher priority relative to QoS rules to more efficiently drop packets that match those associated with a virus, worm, or DoS attack.

For the purposes of classifying IP protocols, the following objects listed in the second column are encoded within TLV 60 and shown in comparison with [RFC 2669] to construct L3/L4 rule criteria to enforce the operator's security policy.

Table 79 - Mapping of docsDevFilterIpTable (RFC 2669) to UDCs for Layer 3 & 4 Criteria

IP Filters [RFC 2669]	UDC TLV 60 Encodings	Description
docsDevFilterIpIndex	Id	Rule index
docsDevFilterIpControl	- no equivalent	discard, accept, policy(*1)
docsDevFilterIpIfIndex	CMIM	CM interface(s)(*2)
docsDevFilterIpDirection	- no equivalent	inbound, outbound, both(*3)
docsDevFilterIpBroadcast	- no equivalent	Broadcast and multicast or all packets
- no equivalent	Rule priority	Directs order of processing
docsDevFilterIpStatus	- no equivalent	Activation state(*4)
docsDevFilterIpProtocol	IpProtocol	IP transport type, e.g., TCP, UDP
- no equivalent	FlowLabel	IPv6 flow label
docsDevFilterIpSaddr	IpSourceAddr	IP source address/prefix
docsDevFilterIpSmask	IpSourceMask	IP source mask/prefix length
docsDevFilterIpDaddr	IpDestAddr	IP dest. Address/prefix
docsDevFilterIpDmask	IpDestMask	IP dest. mask/prefix length
docsDevFilterIpTos	IpTosLow	Legacy type of service range low
	IpTosHigh	Legacy type of service range high
docsDevFilterIpTosMask	IpTosMask	Legacy type of service mask
docsDevFilterIpSourcePortLow	SourcePortStart	TCP/UDP source port range start
docsDevFilterIpSourcePortHigh	SourcePortEnd	TCP/UDP source port range end
docsDevFilterIpDestPortLow	DestPortStart	TCP/UDP source port range start
docsDevFilterIpDestPortHigh	DestPortEnd	TCP/UDP source port range end
docsDevFilterIpContinue	- no equivalent	Continue comparing rules on matches(*5)
docsDevFilterIpPolicyId	- no equivalent	Extensions for other criterion

TABLE NOTES:
(*1) UDCs only perform discard actions.
(*2) CMIM allows for multiple interfaces per rule, while [RFC 2669] aggregates only CPE interface.
(*3) UDCs only perform upstream filtering.
(*4) UDCs are always active. The SNMP docsDevFilterIpTable table provides RowStatus for controlling the activation state of IP filters.
(*5) UDCs do not continue performing packet comparisons after a match.

The SNMP table docsQosPktClassTable from [DOCS-QOS3-MIB] is used for reporting of both QoS Classifiers and Drop Classifiers at the CM. The docsQosPktClassPkts object within docsQosPktClassTable is used to count packet matches to each classifier rule.

B.1.4.2 LLC/MAC Classification through UDCs

L2 criteria such as MAC address source and destination, header type, 802.1p/q VLAN tag or user_priority and Cable Modem Interface Mask (CMIM) may be used to classify packets as deemed necessary by the operator.

For the purposes of classifying MAC protocols, the following variables listed in the second column are encoded within TLV 60 and shown in comparison with [RFC 2669]. The variables described here are used to construct L2 rule criteria to enforce the operator's security policy.

Table 80 - Upstream Drop Classification Values for LLC/MAC Classification

LLC Filters [RFC 2669]	UDC TLV 60 Encodings	Description
docsDevFilterLLCIndex	Id	Rule index
docsDevFilterLLCIfIndex	CMIM	CM interface
- no equivalent	Rule priority	Directs order of processing
docsDevFilterLLCStatus	- no equivalent	Activation state
- no equivalent	SourceMacAddr	Source MAC address
- no equivalent	DestMacAddr	Destination MAC address
docsDevFilterLLCProtocolType	EnetProtocolType	Ethernet protocol type
docsDevFilterLLCProtocol	EnetProtocol	Ethernet protocol
- no equivalent	802.1p User priority low	Ethernet user priority range low
- no equivalent	802.1p User priority high	Ethernet user priority range high
- no equivalent	VLAN ID	12-bit Ethernet VLAN ID

The SNMP table docsQosPktClassTable from DOCS-QOS3-MIB is used for reporting of both QoS Classifiers and Drop Classifiers at the CM. The docsQosPktClassPkts object within docsQosPktClassTable is used to count packet matches to each classifier rule.

Annex C Format and Content for Event, SYSLOG, and SNMP Notification (Normative)

The CM MUST format the CM MAC Address field <CM-MAC> of the Event Message text, including such instances of docsDevEvText, using lowercase letters.

The CM MUST format the CMTS MAC Address field <CMTS-MAC> of the Event Message text, including such instances of docsDevEvText, using lower case letters.

The CM MAY append additional vendor-specific text to the end of the event text reported in the docsDevEvText object and the syslog text field.

Table 81 in this annex summarizes the format and content for event, syslog, and SNMP notifications required for a DOCSIS 4.0-compliant CM.

Each row specifies a possible event that may appear in the CM. These events are to be reported by a CM through local event logging, and may be accompanied by syslog or SNMP notification.

The "Process" and "Sub-Process" columns indicate in which stage the event happens. The "CM Priority" column indicates the priority the event is assigned in the CM. The priority is the same as is reported in the docsDevEvLevel object in the cable device MIB [RFC 4639] and in the LEVEL field of the syslog.

The "Event Message" column specifies the event text, which is reported in the docsDevEvText object of the cable device MIB and the text field of the syslog. The "Message Notes And Details" column provides additional information about the event text in the "Event Message" column. Some of the text fields include variable information. The variables are explained in the "Message Notes And Details" column. For events where the "Event Message" or "Message Notes and Details" column includes either <P1> or <P2>, there is a single space between the value as defined by the <P1> or <P2> and the preceding text.

Example SNMP Notification and Syslog message "Event Message" text string for Event ID 69020900:

SNMP CVC Validation Failure SNMP Manager: 10.50.1.11;CM-MAC=00:22:ce:03:f4:da;CMTS-MAC=00:15:20:00:25:ab;CM-QOS=1.1;CM-VER=3.0;

This specification defines the following keywords as part of the "Event Message" column:

"<TAGS>" (without the quotes) corresponds to:

```
For the CM (without the quotes):        ";<CM-MAC>;<CMTS-MAC>;<CM-QOS>;<CM-VER>;"
```

Where:

```
<CM-MAC>:     CM MAC Address;

Format*: "CM-MAC=xx:xx:xx:xx:xx:xx"

<CMTS-MAC>:   CMTS MAC Address;

Format*: "CMTS-MAC=xx:xx:xx:xx:xx:xx"

<CM-QOS>:     CM DOCSIS QOS Version;

Format*: "CM-QOS=1.0" or "CM-QOS=1.1"

<CM-VER>:     CM DOCSIS Version;

Format*: "CM-VER=1.1" or "CM-VER=2.0" or "CM-VER=3.0" or "CM-VER=3.1" or "CM-VER=4.0"

(*) without the quotes
```

Example SNMP Notification and Syslog message "Event Message" text string for Event ID 69010100:

```
SW Download INIT - Via NMS SW file: junk.bin - SW server: 10.50.1.11;CM-
MAC=00:22:ce:03:f4:da;CMTS-MAC=00:15:20:00:25:ab;CM-QOS=1.1;CM-VER=3.0;
```

The "Error Code Set" column specifies the error code. The "Event ID" column indicates a unique identification number for the event, which is assigned to the docsDevEvId object in the cable device MIB and the <eventId> field of the syslog. The "Notification Name" column specifies the SNMP notification, which notifies this event to an SNMP notification receiver.

The syslog format, as well as the rules to uniquely generate an event ID from the error code, are described in Section 8.1.2.1.3 of this specification.

Table 81 - Event Format and Content

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Authentication and Encryption							
			<Reserved>			0	
BPKM	AUTH-FSM	Warning	Auth Reject - No Information<TAGS>		B301.2	66030102	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Reject - Unauthorized CM<TAGS>		B301.3	66030103	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Reject - Unauthorized SAID<TAGS>		B301.4	66030104	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Error	Auth Reject - Permanent Authorization Failure<TAGS>		B301.8	66030108	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Reject - Time of Day not acquired<TAGS>		B301.9	66030109	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Informational	Auth Reject - EAE disabled<TAGS>		B301.10	66030110	
BPKM	AUTH-FSM	Alert	CM Certificate Error<TAGS>		B301.11	66030111	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Invalid - No Information<TAGS>		B302.2	66030202	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Invalid - Unauthorized CM<TAGS>		B302.3	66030203	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Invalid - Unsolicited<TAGS>		B302.5	66030205	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Invalid - Invalid Key Sequence Number<TAGS>		B302.6	66030206	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Auth Invalid - Message (Key Request) Authentication Failure<TAGS>		B302.7	66030207	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Warning	Unsupported Crypto Suite<TAGS>		B303.0	66030300	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Informational	Authorized<TAGS>		B401.0	66040100	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Informational	Auth Pend<TAGS>		B402.0	66040200	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Informational	Auth Comp<TAGS>		B403.0	66040300	CM: docsIf3CmEventNotif
BPKM	AUTH-FSM	Informational	Stop<TAGS>		B404.0	66040400	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
BPKM	TEK-FSM	Warning	Key Reject - No Information<TAGS>		B501.2	66050102	CM: docsIf3CmEventNotif
BPKM	TEK-FSM	Warning	Key Reject - Unauthorized SAID<TAGS>		B501.3	66050103	CM: docsIf3CmEventNotif
BPKM	TEK-FSM	Warning	TEK Invalid - No Information<TAGS>		B502.3	66050203	CM: docsIf3CmEventNotif
BPKM	TEK-FSM	Warning	TEK Invalid - Invalid Key Sequence Number<TAGS>		B502.6	66050206	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Informational	SA Map State Machine Started<TAGS>		B601.0	66060100	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Warning	Unsupported Crypto Suite<TAGS>		B602.0	66060200	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Error	Map Request Retry Timeout<TAGS>		B603.0	66060300	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Informational	Unmap<TAGS>		B604.0	66060400	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Informational	Map Reject - Downstream Traffic Flow Not Mapped to BPI+ SAID (EC=8)<TAGS>		B605.10	66060510	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Warning	Map Reject - Not Authorized for Requested Downstream Traffic Flow (EC=7)<TAGS>		B605.9	66060509	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Warning	Mapped to Existing SAID<TAGS>		B606.0	66060600	CM: docsIf3CmEventNotif
Dynamic SA	SA MAP-FSM	Warning	Mapped to New SAID<TAGS>		B607.0	66060700	CM: docsIf3CmEventNotif
DBC and DCC							
DBC	DBC Request	Warning	CMTS Bad DBC - confirmation code <P1>: <P2><TAGS>	P1=<Confirmation Code> P2=<Confirmation> See Annex C.4 Confirmation Code	C501.0	67050100	
DBC	DBC Request	Warning	DBC- denied - confirmation code <P1>: <P2><TAGS>	P1=<Confirmation Code> P2=<Confirmation> See Annex C.4 Confirmation Code	C502.0	67050200	
DBC	DBC Request	Warning	DBC-REQ Mismatch Between Calculated Value for P1.6hi Compared to CCAP Provided Value<TAGS>		C503.0	67050300	
DBC	DBC Acknowledgement	Error	DBC-ACK not received<TAGS>		C701.0	67070100	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DBC	DBC Acknowledgement	Notice	Bad CMTS DBC-ACK: <P1><TAGS>	P1="unspecified reason" | "unknown transaction ID" | "authentication failure" | "msg syntax error"	C702.0	67070200	
DCC	DCC Request	Error	DCC rejected already there<TAGS>		C201.0	67020100	CM: docsIf3CmEventNotif
DCC	DCC Request	Informational	DCC depart old<TAGS>		C202.0	67020200	CM: docsIf3CmEventNotif
DCC	DCC Request	Informational	DCC arrive new<TAGS>		C203.0	67020300	CM: docsIf3CmEventNotif
DCC	DCC Request	Critical	DCC aborted unable to acquire new downstream channel<TAGS>		C204.0	67020400	
DCC	DCC Request	Critical	DCC aborted no UCD for new upstream channel<TAGS>		C205.0	67020500	
DCC	DCC Request	Critical	DCC aborted unable to communicate on new upstream channel<TAGS>		C206.0	67020600	
DCC	DCC Request	Error	DCC rejected unspecified reason<TAGS>		C207.0	67020700	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected permanent - DCC not supported<TAGS>		C208.0	67020800	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected service flow not found<TAGS>		C209.0	67020900	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected required parameter not present<TAGS>		C210.0	67021000	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected authentication failure<TAGS>		C211.0	67021100	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected multiple errors<TAGS>		C212.0	67021200	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected, duplicate SF reference-ID or index in message<TAGS>		C215.0	67021500	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected parameter invalid for context<TAGS>		C216.0	67021600	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected message syntax error<TAGS>		C217.0	67021700	CM: docsIf3CmEventNotif
DCC	DCC Request	Error	DCC rejected message too big<TAGS>		C218.0	67021800	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DCC	DCC Request	Error	DCC rejected 2.0 mode disabled<TAGS>		C219.0	67021900	CM: docsIf3CmEventNotif
DCC	DCC Acknowledgement	Error	DCC-ACK not received<TAGS>		C401.0	67040100	CM: docsIf3CmEventNotif
DCC	DCC Acknowledgement	Error	DCC-ACK rejected unspecified reason<TAGS>		C402.0	67040200	CM: docsIf3CmEventNotif
DCC	DCC Acknowledgement	Error	DCC-ACK rejected unknown transaction ID<TAGS>		C403.0	67040300	CM: docsIf3CmEventNotif
DCC	DCC Acknowledgement	Error	DCC-ACK rejected authentication failure<TAGS>		C404.0	67040400	CM: docsIf3CmEventNotif
DCC	DCC Acknowledgement	Error	DCC-ACK rejected message syntax error<TAGS>		C405.0	67040500	CM: docsIf3CmEventNotif
DHCP, TOD and TFTP							
DHCP		Error	DHCP RENEW sent - No response for <P1><TAGS>	P1=IPv4 or IPv6	D101.0	68010100	
DHCP		Error	DHCP REBIND sent - No response for <P1><TAGS>	P1=IPv4 or IPv6	D102.0	68010200	
DHCP		Error	DHCP RENEW WARNING - Field invalid in response <P1> option<TAGS>	P1=v4	D103.0	68010300	
DHCP		Critical	DHCP RENEW FAILED - Critical field invalid in response		D103.1	68010301	
DHCP		Error	DHCP REBIND WARNING - Field invalid in response <TAGS>		D104.0	68010400	
DHCP		Critical	DHCP REBIND FAILED - Critical field invalid in response		D104.1	68010401	
DHCP		Notice	DHCP Reconfigure received<TAGS>		D105.0	68010500	
DHCP		Notice	DHCP Renew - lease parameters <P1> modified<TAGS>	P1 = list of params that changed at renew	D106.0	68010600	
DHCP		Error	Primary lease failed, IPv4 fallback initiated<TAGS>		D107.0	68010700	
Init	DHCP	Critical	DHCP FAILED - Discover sent, no offer received<TAGS>		D01.0	68000100	
Init	DHCP	Critical	DHCP FAILED - Request sent, No response<TAGS>		D02.0	68000200	
Init	DHCP	Warning	DHCP WARNING - Non-critical field invalid in response <TAGS>		D03.0	68000300	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Init	DHCP	Critical	DHCP FAILED - Critical field invalid in response <TAGS>		D03.1	68000301	
Init	DHCP	Critical	DHCP failed - RS sent, no RA received<TAGS>		D12.0	68001200	
Init	DHCP	Critical	DHCP Failed - Invalid RA<TAGS>		D12.1	68001201	
Init	DHCP	Critical	DHCP failed - DHCP Solicit sent, No DHCP Advertise received<TAGS>		D12.2	68001202	
Init	DHCP	Critical	DHCP failed - DHCP Request sent, No DHCP REPLY received<TAGS>		D12.3	68001203	
Init	DHCP	Error	Primary address acquired, secondary failed<TAGS>		D12.4	68001204	
Init	DHCP	Error	Primary address failed, secondary active<TAGS>		D12.5	68001205	
Init	IPv6 Address Acquisition	Critical	Link-Local address failed DAD<TAGS>		D13.1	68001301	
Init	IPv6 Address Acquisition	Critical	DHCP lease address failed DAD<TAGS>		D13.2	68001302	
Init	TOD	Warning	ToD request sent - No Response received<TAGS>		D04.1	68000401	
Init	TOD	Warning	ToD Response received - Invalid data format<TAGS>		D04.2	68000402	
Init	TFTP	Critical	TFTP failed - Request sent - No Response<TAGS>		D05.0	68000500	
Init	TFTP	Critical	TFTP failed - configuration file NOT FOUND<TAGS>	For SYSLOG only: append: File name = <P1> P1 = requested file name	D06.0	68000600	
Init	TFTP	Critical	TFTP Failed - OUT OF ORDER packets<TAGS>		D07.0	68000700	
Init	TFTP	Critical	TFTP file complete - but failed Message Integrity check MIC<TAGS>	For SYSLOG only: append: File name = <P1> P1 = file name of TFTP file	D08.0	68000800	
Init	TFTP	Critical	TFTP file complete - but missing mandatory TLV<TAGS>		D09.0	68000900	
Init	TFTP	Critical	TFTP Failed - file too big<TAGS>		D10.0	68001000	
Init	TFTP	Critical	TFTP Request Retries exceeded, CM unable to register	For SYSLOG only: append: File name = <P1> P1 = file name of TFTP file	D11.1	68001101	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Init	TFTP	Critical	Config File Rejected due to Invalid or Unexpected TLV <P1> encoding<TAGS>	P1 = First TLV encoding (e.g., 53, 54, 60, etc.) which caused cfg file rejection	D11.2	68001102	CM: docsIf3CmEventNotif
TOD		Error	ToD request sent - No Response received<TAGS>		D04.3	68000403	CM: docsIf3CmEventNotif
TOD		Error	ToD Response received - Invalid data format<TAGS>		D04.4	68000404	CM: docsIf3CmEventNotif
SSH Services							
SSH SERVICES	SSH INIT	Information	SSH INIT via config file; Enabled Interface: <P1>; Connection Timeout: <P2>; <TAGS>;	P1 = EnabledInterfaces of the SshServer object P2 = NewConnectionTimeout of the SshServer object	D500.00	68050000	
SSH SERVICES	SSH OPERATIONS	Information	SSH service disabled for all new connections, new connections timeout expired; <TAGS>;		D501.00	68050100	
SSH SERVICES	SSH OPERATIONS	Notice	SSH connection rejected, user credential error; User: <P1>; Credential Type: <P2>; <TAGS>;	P1 = SSH Username P2 = User Credential Type	D501.01	68050001	
SSH SERVICES	SSH OPERATIONS	Information	SSH connection successful via user login; User: <P1>; Credential Type: <P2>; <TAGS>;	P1 = SSH Username P2 = User Credential Type	D501.02	68050002	
SSH SERVICES	SSH OPERATIONS	Information	Existing SSH connection closed, inactivity timeout expired; User: <P1>; <TAGS>;	P1 = SSH Username	D501.03	68050003	
SSH SERVICES	SSH GENERAL FAILURE	Warning	SSH connection rejected, SSH protocol failure; User: <P1>; SSH Error: <P2>; <TAGS>;	P1 = SSH Username P2 = SSH Error from Table 85	D502.00	68050200	
SSH SERVICES	SSH Remote AUTH Failure	Notice	SSH connection rejected, server URL misconfiguration; Auth Server: <P1>; User: <P2>; Config File: <P3>; <TAGS>;	P1 = SCCA REST API URL P2 = SSH Username P3 = Config File Name	D503.00	68050300	
SSH SERVICES	SSH Remote AUTH Failure	Notice	SSH connection rejected, TLS server certificate error; Auth Server: <P1>; User: <P2>; Cert Error: <P3>; Cert Level: <P4>; <TAGS>;	P1 = Server Address P2 = SSH Username P3 = Certificate Error from Table 83 P4 = Certificate Level from Table 82	D503.01	68050301	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
SSH SERVICES	SSH Remote AUTH Failure	Notice	SSH connection rejected, TLS protocol error; Auth Server: <P1>; User: <P2>; TLS Error: <P3>; Retry Number: <P4>; <TAGS>;	P1 = Server Address P2 = SSH Username P3 = TLS Error from Table 84 P4 = Maximum Retry number (default: 3)	D503.02	68050302	
SSH SERVICES	SSH CDS Download	Warning	BPI+ EAE not used; Config File: <P1>; <TAGS>;	P1 = Config File Name	D504.00	68050400	
SSH SERVICES	SSH CDS Download	Notice	SSH connection rejected, server URL misconfiguration; CDS Server: <P1>; Config File: <P2>; <TAGS>;	P1 = CDS Download URL P2 = Config File Name	D504.01	68050401	
SSH SERVICES	SSH CDS Download	Notice	SSH connection rejected, TLS server certificate error; CDS Server: <P1>; Cert Error: <P2>; Cert Level: <P3>; <TAGS>;	P1 = Server Address P2 = Certificate Error from Table 83 P3 = Certificate Level from Table 82	D504.02	68050402	
SSH SERVICES	SSH CDS Download	Notice	SSH connection rejected, TLS protocol error; CDS Server: <P1>; TLS Error: <P2>; Retry Number: <P3>; <TAGS>;	P1 = Server Address P2 = TLS Error from Table 84 P3 = Maximum Retry number (default: 3)	D504.03	68050403	
SSH SERVICES	SSH CDS Download	Notice	CDS has been successfully downloaded; CDS: <P1>; CDS Server: <P2>; <TAGS>;	P1 = CDS Name P2 = SSH CDS Download URL	D504.04	68050404	
SSH SERVICES	SSH CDS Download	Error	Authentication failure, unable to decrypt CDS; CDS: <P1>; CDS Server: <P2>; <TAGS>;	P1 = CDS Name P2 = SSH CDS Download URL	D504.05	68050405	docsIf3CmEventNotif
SSH SERVICES	SSH CDS Download	Error	Authentication failure, failed to verify CDS signature; CDS: <P1>; CDS Server: <P2>; <TAGS>;	P1 = CDS Name P2 = SSH CDS Download URL	D504.06	68050406	docsIf3CmEventNotif
SSH SERVICES	SSH CDS Download	Error	Authentication failure, unable to install user credentials; CDS: <P1>; CDS Server: <P2>; <TAGS>;	P1 = CDS Name P2 = SSH CDS Download URL	D504.07	68050407	docsIf3CmEventNotif
SSH SERVICES	SSH CDS Download	Notice	CDS has been successfully removed; <TAGS>;		D504.08	68050408	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Secure Software Download							
SW Upgrade	SW UPGRADE INIT	Notice	SW Download INIT - Via NMS	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E101.0	69010100	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE INIT	Notice	SW Download INIT - Via Config file <P1>	Other than Local Log, append: SW file: <P2> - SW server: < P3><TAGS> P1 = CM config file name P2 = SW file name P3 = SW Download server IP address	E102.0	69010200	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	SW Upgrade Failed during download - Max retry exceed (3)	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E103.0	69010300	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	SW Upgrade Failed Before Download - Server not Present	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E104.0	69010400	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	SW upgrade Failed before download - File not Present	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E105.0	69010500	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	SW upgrade Failed before download - TFTP Max Retry Exceeded	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E106.0	69010600	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	SW upgrade Failed after download - Incompatible SW file	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E107.0	69010700	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	SW upgrade Failed after download - SW File corruption	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E108.0	69010800	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Disruption during SW download - Power Failure	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E109.0	69010900	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Disruption during SW download - RF removed	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E110.0	69011000	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
SW Upgrade	SW UPGRADE SUCCESS	Notice	SW download Successful - Via NMS	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E111.0	69011100	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE SUCCESS	Notice	SW download Successful - Via Config file	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E112.0	69011200	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Improper Code File Controls	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E201.0	69020100	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Code File Manufacturer CVC Validation Failure	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E202.0	69020200	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Code File Manufacturer CVS Validation Failure	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E203.0	69020300	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Code File Co-Signer CVC Validation Failure	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E204.0	69020400	CM: docsIf3CmEventNotif
SW Upgrade	SW UPGRADE GENERAL FAILURE	Error	Code File Co-Signer CVS Validation Failure	Other than Local Log, append: SW file: <P1> - SW server: < P2><TAGS> P1 = SW file name P2 = SW Download server IP address	E205.0	69020500	CM: docsIf3CmEventNotif
SW Upgrade	VERIFICATION OF CVC	Error	Improper Configuration File CVC Format	Other than Local Log, append: Config file: <P1> - Config file server: < P2><TAGS> P1 = Config file name P2 = Config file server IP address	E206.0	69020600	CM: docsIf3CmEventNotif
SW Upgrade	VERIFICATION OF CVC	Error	Configuration File CVC Validation Failure	Other than Local Log, append: Config file: <P1> - Config file server: < P2><TAGS> P1 = Config file name P2 = Config file server IP address	E207.0	69020700	CM: docsIf3CmEventNotif
SW Upgrade	VERIFICATION OF CVC	Error	Improper SNMP CVC Format	Other than local Log, append: SNMP Manager: <P1><TAGS> P1= IP Address of SNMP Manager	E208.0	69020800	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
SW Upgrade	VERIFICATION OF CVC	Error	SNMP CVC Validation Failure	Other than local Log, append: SNMP Manager: <P1><TAGS> P1= IP Address of SNMP Manager	E209.0	69020900	CM: docsIf3CmEventNotif
Registration and TLV-11							
Init	REGISTRATION RESPONSE	Critical	REG-RSP - invalid format or not recognized;<TAGS>		I01.0	73000100	
Init	REGISTRATION RESPONSE	Critical	REG RSP not received<TAGS>		I02.0	73000200	
Init	REGISTRATION RESPONSE	Critical	REG RSP bad SID <P1><TAGS>	P1 = SID which was reject	I03.0	73000300	
Init	1.1 and 2.0 SPECIFIC REGISTRATION RESPONSE	Critical	REG RSP contains service flow parameters that CM cannot support <P1><TAGS>	P1 = Service Flow ID	I251.0	73025100	
Init	1.1 and 2.0 SPECIFIC REGISTRATION RESPONSE	Critical	REG RSP contains classifier parameters that CM cannot support <P1><TAGS>	P1 = Service Flow ID	I251.1	73025101	
Init	1.1 and 2.0 SPECIFIC REGISTRATION RESPONSE	Critical	Registration RSP rejected unspecified reason<TAGS>		I251.3	73025103	
Init	1.1 and 2.0 SPECIFIC REGISTRATION RESPONSE	Critical	Registration RSP rejected message syntax error <P1><TAGS>	P1 = message	I251.4	73025104	
Init	1.1 and 2.0 SPECIFIC REGISTRATION RESPONSE	Critical	Registration RSP rejected message too big <P1><TAGS>	P1 = # of characters	I251.5	73025105	
Init	2.0 SPECIFIC REGISTRATION RESPONSE	Warning	REG-RSP received after REG-ACK. Returning to 1.x transmit mode<TAGS>		I261.0	73026100	
Init	TLV-11 PARSING	Notice	TLV-11 - unrecognized OID<TAGS>		I401.0	73040100	CM: docsIf3CmEventNotif
Init	TLV-11 PARSING	Critical	TLV-11 - Illegal Set operation failed<TAGS>		I402.0	73040200	CM: docsIf3CmEventNotif
Init	TLV-11 PARSING	Critical	TLV-11 - Failed to set duplicate elements<TAGS>		I403.0	73040300	CM: docsIf3CmEventNotif
Init	Waiting for REG-RSP or REG-RSP-MP	Error	T6 Timeout and retries exceeded<TAGS>		I271.0	73027100	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Init	CM Complete Registration	Error	Cannot create US Primary Service Flow<TAGS>		I501.0	73050100	
Init	CM Complete Registration	Notice	Received REG-RSP while in REG-HOLD1 state<TAGS>		I502.0	73050200	
Init	CM Complete Registration	Notice	Received REG-RSP while in REG-HOLD2 state<TAGS>		I503.0	73050300	
Init	CM Complete Registration	Warning	REG-RSP-MP Mismatch Between Calculated Value for P1.6hi Compared to CCAP Provided Value<TAGS>		I504.0	73050400	
General							
Init		Informational	A transmit opportunity was missed because the MAP arrived too late.		N01.0	78000100	
Ranging							
Init	RANGING	Critical	No Maintenance Broadcasts for Ranging opportunities received - T2 time-out<TAGS>		R01.0	82000100	
Init	RANGING	Critical	No Ranging Response received - T3 time-out<TAGS>		R02.0	82000200	
Init	RANGING	Critical	Ranging Request Retries exhausted<TAGS>		R03.0	82000300	
Init	RANGING	Critical	Received Response to Broadcast Maintenance Request, But no Unicast Maintenance opportunities received - T4 time out<TAGS>		R04.0	82000400	
Init	RANGING	Critical	Started Unicast Maintenance Ranging - No Response received - T3 time-out<TAGS>		R05.0	82000500	
Init	RANGING	Critical	Unicast Maintenance Ranging attempted - No response - Retries exhausted<TAGS>		R06.0	82000600	
Init	RANGING	Critical	Unicast Ranging Received Abort Response - Re-initializing MAC<TAGS>		R07.0	82000700	
Init	RANGING	Critical	16 consecutive T3 timeouts while trying to range on upstream channel <P1><TAGS>	P1 = Upstream Channel ID	R08.0	82000800	
Init	RANGING	Warning	B-INIT-RNG Failure - Retries exceeded<TAGS>		R09.0	82000900	
Station Maintenance	RANGING	Warning	RNG-RSP CCAP Commanded Power Exceeds Value for Pmax<TAGS>		R10.0	82001000	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Station Maintenance	RANGING	Warning	RNG-RSP CCAP Commanded Power Exceeds Value Corresponding to the Top of the DRW<TAGS>		R11.0	82001100	
Station Maintenance	RANGING	Warning	RNG-RSP CCAP Commanded Power in Excess of 6 dB Below the Value Corresponding to the Top of the DRW<TAGS>		R12.0	82001200	
Dynamic Services							
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Unspecified reason<TAGS>		S01.0	83000100	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Unrecognized configuration setting<TAGS>		S01.1	83000101	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Classifier not found<TAGS>		S01.10	83000110	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Classifier exists<TAGS>		S01.11	83000111	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Duplicated reference-ID or index in message<TAGS>		S01.14	83000114	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Multiple upstream flows<TAGS>		S01.15	83000115	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Multiple downstream flows<TAGS>		S01.16	83000116	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Classifier for another flow<TAGS>		S01.17	83000117	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Parameter invalid for context<TAGS>		S01.19	83000119	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Temporary no resource<TAGS>		S01.2	83000102	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Authorization failure<TAGS>		S01.20	83000120	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Major service flow error<TAGS>		S01.21	83000121	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Major classifier error<TAGS>		S01.22	83000122	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Multiple major errors<TAGS>		S01.24	83000124	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Message syntax error<TAGS>		S01.25	83000125	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Message too big<TAGS>		S01.26	83000126	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Temporary DCC<TAGS>		S01.27	83000127	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Permanent administrative<TAGS>		S01.3	83000103	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Required parameter not present<TAGS>		S01.4	83000104	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Header suppression setting not supported<TAGS>		S01.5	83000105	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Service flow exists<TAGS>		S01.6	83000106	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - HMAC Auth failure<TAGS>		S01.7	83000107	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Add aborted<TAGS>		S01.8	83000108	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Add rejected - Multiple errors<TAGS>		S01.9	83000109	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Unspecified reason<TAGS>		S02.0	83000200	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Unrecognized configuration setting<TAGS>		S02.1	83000201	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Classifier not found<TAGS>		S02.10	83000210	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Classifier exists<TAGS>		S02.11	83000211	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Duplicated reference-ID or index in message<TAGS>		S02.14	83000214	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Multiple upstream flows<TAGS>		S02.15	83000215	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Multiple downstream flows<TAGS>		S02.16	83000216	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Classifier for another flow<TAGS>		S02.17	83000217	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Invalid parameter for context<TAGS>		S02.19	83000219	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Temporary no resource<TAGS>		S02.2	83000202	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Authorization failure<TAGS>		S02.20	83000220	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Major service flow error<TAGS>		S02.21	83000221	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Major classifier error<TAGS>		S02.22	83000222	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Multiple major errors<TAGS>		S02.24	83000224	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Message syntax error<TAGS>		S02.25	83000225	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Message too big<TAGS>		S02.26	83000226	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Temporary DCC<TAGS>		S02.27	83000227	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Permanent administrative<TAGS>		S02.3	83000203	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Requester not owner of service flow<TAGS>		S02.4	83000204	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Service flow not found<TAGS>		S02.5	83000205	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Required parameter not present<TAGS>		S02.6	83000206	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Header suppression setting not supported<TAGS>		S02.7	83000207	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - HMAC Auth failure<TAGS>		S02.8	83000208	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Change rejected - Multiple errors<TAGS>		S02.9	83000209	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Delete rejected - Unspecified reason<TAGS>		S03.0	83000300	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Delete rejected -Requester not owner of service flow<TAGS>		S03.1	83000301	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Delete rejected - Service flow not found<TAGS>		S03.2	83000302	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Delete rejected - HMAC Auth failure<TAGS>		S03.3	83000303	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE REQUEST	Error	Service Delete rejected - Message syntax error<TAGS>		S03.4	83000304	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Invalid transaction ID<TAGS>		S101.0	83010100	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add aborted - No RSP<TAGS>		S101.1	83010101	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Duplicate reference_ID or index in message<TAGS>		S101.11	83010111	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Classifier for another flow<TAGS>		S101.12	83010112	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Parameter invalid for context<TAGS>		S101.13	83010113	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Major service flow error<TAGS>		S101.14	83010114	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Major classifier error<TAGS>		S101.15	83010115	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Multiple major errors<TAGS>		S101.17	83010117	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Message too big<TAGS>		S101.18	83010118	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - HMAC Auth failure<TAGS>		S101.2	83010102	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Message syntax error<TAGS>		S101.3	83010103	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Unspecified reason<TAGS>		S101.4	83010104	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Unrecognized configuration setting<TAGS>		S101.5	83010105	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Required parameter not present<TAGS>		S101.6	83010106	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Service Flow exists<TAGS>		S101.7	83010107	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Multiple errors<TAGS>		S101.8	83010108	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Add Response rejected - Classifier exists<TAGS>		S101.9	83010109	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Invalid transaction ID<TAGS>		S102.0	83010200	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change aborted - No RSP<TAGS>		S102.1	83010201	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Duplicated reference-ID or index in<TAGS>		S102.10	83010210	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Invalid parameter for context<TAGS>		S102.11	83010211	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Major classifier error<TAGS>		S102.12	83010212	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Multiple Major errors<TAGS>		S102.14	83010214	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Message too big<TAGS>		S102.15	83010215	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - HMAC Auth failure<TAGS>		S102.2	83010202	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Message syntax error<TAGS>		S102.3	83010203	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Unspecified reason<TAGS>		S102.4	83010204	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Unrecognized configuration setting<TAGS>		S102.5	83010205	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Required parameter not present<TAGS>		S102.6	83010206	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Multiple errors<TAGS>		S102.7	83010207	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Change Response rejected - Classifier exists<TAGS>		S102.8	83010208	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE RESPONSE	Error	Service Delete Response rejected - Invalid transaction ID<TAGS>		S103.0	83010300	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Add Response rejected - Invalid Transaction ID<TAGS>		S201.0	83020100	CM: docsIf3CmEventNotif

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Add Aborted - No ACK<TAGS>		S201.1	83020101	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Add ACK rejected - HMAC auth failure<TAGS>		S201.2	83020102	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Add ACK rejected - Message syntax error<TAGS>		S201.3	83020103	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Change ACK rejected - Invalid transaction ID<TAGS>		S202.0	83020200	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Change Aborted - No ACK<TAGS>		S202.1	83020201	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Change ACK rejected - HMAC Auth failure<TAGS>		S202.2	83020202	CM: docsIf3CmEventNotif
DYNAMIC SERVICES	DYNAMIC SERVICE ACKNOWLEDGEMENT	Error	Service Change ACK rejected - Message syntax error<TAGS>		S202.3	83020203	CM: docsIf3CmEventNotif
Downstream Acquisition							
Init	DOWNSTREAM ACQUISITION	Critical	SYNC Timing Synchronization failure - Failed to acquire QAM/QPSK symbol timing;<TAGS>		T01.0	84000100	
Init	DOWNSTREAM ACQUISITION	Critical	SYNC Timing Synchronization failure - Failed to acquire FEC framing<TAGS>		T02.0	84000200	
Init	DOWNSTREAM ACQUISITION	Critical	SYNC Timing Synchronization failure, Acquired FEC framing - Failed to acquire MPEG2 Sync<TAGS>		T02.1	84000201	
Init	DOWNSTREAM ACQUISITION	Critical	SYNC Timing Synchronization failure - Failed to acquire MAC framing<TAGS>		T03.0	84000300	
Init	DOWNSTREAM ACQUISITION	Critical	SYNC Timing Synchronization failure - Failed to receive MAC SYNC frame within time-out period<TAGS>		T04.0	84000400	
Init	DOWNSTREAM ACQUISITION	Critical	SYNC Timing Synchronization failure - Loss of Sync<TAGS>		T05.0	84000500	
Init	DOWNSTREAM ACQUISITION	Error	RCS Primary DS Failure<TAGS>		T06.0	84000600	
Init	DOWNSTREAM ACQUISITION	Warning	RCS Partial Service<TAGS>		T07.0	84000700	
Init	RCP and RCC	Error	RCP-ID in RCC not supported<TAGS>		T101.0	84010100	
Init	RCP and RCC	Error	More than one RCP-ID included in RCC<TAGS>		T102.0	84010200	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Init	RCP and RCC	Error	Receive Module Index missing in RCC<TAGS>		T103.0	84010300	
Init	RCP and RCC	Error	RCC contains a Receive Module Index which is not supported<TAGS>		T104.0	84010400	
Init	RCP and RCC	Error	Receive channel center frequency not within allowed range of center frequencies for Receive Module<TAGS>		T105.0	84010500	
Init	RCP and RCC	Error	Receive Module first channel center frequency not within allowed range of center frequencies<TAGS>		T106.0	84010600	
Init	RCP and RCC	Error	Receive Module first channel center frequency not present in RCC<TAGS>		T107.0	84010700	
Init	RCP and RCC	Error	No primary downstream channel assignment in RCC<TAGS>		T108.0	84010800	
Init	RCP and RCC	Error	More than one primary downstream channel assignment present in RCC<TAGS>		T109.0	84010900	
Init	RCP and RCC	Error	Receive Module connectivity encoding in RCC Requires configuration not supported<TAGS>		T110.0	84011000	
Init	RCP and RCC	Error	Receive channel index in RCC not supported by CM<TAGS>		T111.0	84011100	
Init	RCP and RCC	Error	Center frequency in RCC not a multiple of 62500 Hz<TAGS>		T112.0	84011200	
Init	MDD	Error	Missing Mandatory MDD TLV on primary DS Channel<TAGS>		T201.0	84020100	
Init	MDD	Warning	Lost MDD Timeout<TAGS>		T202.0	84020200	
Init	MDD	Warning	MDD message timeout<TAGS>		T203.0	84020300	
Init	OBTAIN UPSTREAM PARAMETERS	Critical	No UCDs Received - Timeout;<TAGS>		U01.0	85000100	
Init	OBTAIN UPSTREAM PARAMETERS	Critical	UCD invalid or channel unusable<TAGS>		U02.0	85000200	
Init	OBTAIN UPSTREAM PARAMETERS	Critical	UCD & SYNC valid - NO MAPS for this channel<TAGS>		U04.0	85000400	
Init	OBTAIN UPSTREAM PARAMETERS	Critical	US channel wide parameters not set before Burst Descriptors<TAGS>		U06.0	85000600	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Init	Acquire CM Transmit Channels	Error	TCS Fail on all Upstream Channels<TAGS>		U101.0	85010100	
Init	Acquire CM Transmit Channels	Warning	TCS Partial Service<TAGS>		U102.0	85010200	
Init	Acquire CM Transmit Channels	Warning	Initializing Channel Timeout Expires - Time the CM can perform initial ranging on all upstream channels in the TCS has expired<TAGS>		U103.0	85010300	
CM-CTRL							
CM-CTRL	CM-CTRL	Debug	CM-CTRL - Command: <P1> (if P1= mute Add Interval: <P2> ChannelID: <P3>) (If P1 = forwarding Add Action: <P4>) <TAGS>	P1 = mute, or cmReinit, or forwarding P2= mute interval, Value 0 indicate unmute operation P3= Channel ID or 0 P4 = enable, or disable	L01.0	76000100	CM: docsIf3CmEventNotif
CM-CTRL	CM-CTRL	Debug	CM-CTRL- Invalid message format<TAGS>		L02.0	76000200	CM: docsIf3CmEventNotif
Energy Management							
EM	EM-REQ	Informational	EM-RSP not received<TAGS>		L101.0	76010100	
EM	EM-REQ	Warning	EM-REQ retries exhausted<TAGS>		L102.0	76010200	
EM	EM-REQ	Informational	EM-RSP received, Reject Temporary, deferring for <P1> seconds<TAGS>	<P1> = time to defer (seconds)	L103.0	76010300	
EM	EM-REQ	Warning	EM-RSP received, Reject Permanent<TAGS>		L104.0	76010400	
EM	DBC	Informational	CM entered EM 1x1 mode; Reason: <P1><TAGS>	P1=Unknown, Activity Detection, eSAFE, CMTS Initiated	L113.0	76011300	CM: docsIf3CmEventNotif
EM	DBC	Informational	CM exited EM 1x1 mode<TAGS>		L114.0	76011400	CM: docsIf3CmEventNotif
EM	Activity Detection	Informational	EM 1x1 Activity Detection Threshold crossed; Reason:<P1><TAGS>	P1=Upstream entry, Downstream entry, Upstream exit, Downstream exit	L115.0	76011500	CM: docsIf3CmEventNotif
EM	EM-REQ	Informational	EM-REQ Sent<TAGS>		L116.0	76011600	

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
CM-STATUS							
Status	Status	Notice	CM-STATUS message sent. Event Type Code: <P1>; Chan ID: <P2>; DSID: <P3>; MAC Addr: <P4>; OFDM/OFDMA Profile ID: <P5>.<TAGS>	P1: CM-STATUS event type code. P2: Upstream/Downstream channel ID; "N/A" if not provided for this status message type. P3: DSID; "N/A" if not provided for this status message type. P4: MAC Address; "N/A" if not provided for this status message type. P5: OFDM or OFDMA Profile ID; "N/A" if not provided for this status message type.	J101.0	74010100	CM: docsIf3CmEventNotif
Profile Change							
Profile	Profile Change	Notice	DS profile assignment change. DS Chan ID: <P1>; Previous Profile: <P2>; New Profile: <P3>.<TAGS>	P1: Downstream channel ID P2: Previous OFDM Profile ID P3: New OFDM Profile ID	C616.0	67061600	CM: docsIf3CmEventNotif
Profile	Profile Change	Notice	US profile assignment change. US Chan ID: <P1>; Previous Profile: <P2>; New Profile: <P3>.<TAGS>	P1: Upstream channel ID P2: Previous OFDMA Profile ID P3: New OFDMA Profile ID	C616.1	67061601	CM: docsIf3CmEventNotif
Profile	Profile Update	Info	DS profile config update. DS Chan ID: <P1>.<TAGS>	P1: Downstream channel ID	C616.2	67061602	CM: docsIf3CmEventNotif
Profile	Profile Update	Info	US profile config update. US Chan ID: <P1>.<TAGS>	P1: Upstream channel ID	C616.3	67061603	CM: docsIf3CmEventNotif
RxMER Reporting							
RxMER	Cancelled Request	Error	DBC or Primary Backup interface Index modified		C860.0	86010100	

The following tables contain parameters for security events found in Table 81.

Table 82 - Certificate Level Parameters

Error Strings	Extended Description
End Entity Certificate	The error generated while validating the end entity certificate
Intermediate CA certificate	The error generated while validating the intermediate CA certificate
Root CA certificate	The error generated while validating the root CA certificate

Table 83 - Certificate Error Parameters

Error Strings	Extended Description
Incomplete DOCSIS 4.0 Certificate chain	Certificate is missing
Certificate is expired	Certificate is expired
Certificate is revoked	Certificate is revoked via CRL
	Certificate is revoked via OCSP
Cannot fetch revocation info	Missing URL in certificate or configuration
	Misconfigured URL or network unavailability
Certificate has Invalid format	Certificate has improper format
Certificate is not trusted	Certificate does not chain to a valid trusted root
Domain mismatch	The certificate does not contain the FQDN or the server IP address
Certificate public key algorithm is not supported	The algorithm of the public key inside this certificate is not supported

Table 84 - TLS Protocol Error Parameters

Error Strings	Extended Description
TLS version incompatibility	SSH client and server use incompatible TLS version
TLS cipher suites incompatibility	SSH client and server use incompatible cipher suites

Table 85 - SSH Protocol Error Parameters

Error Strings	Extended Description
SSH version incompatibility	SSH client and server use incompatible SSH version
SSH cipher suites incompatibility	SSH client and server use incompatible cipher suites

C.1 Deprecated Events

Table 86 in this annex lists deprecated events, including any associated syslog and SNMP trap notifications for the events, for a DOCSIS 4.0-compliant CM. Implementation of deprecated events is optional.

Table 86 - Deprecated Events

Process	Sub-Process	CM Priority	Event Message	Message Notes and Detail	Error Code Set	Event ID	Notification Name
Authentication and Encryption							
Init (BPI+)	DOCSIS 1.0 CONFIG FILE	Error	Missing BP Configuration Setting TLV Type: <P1><TAGS>	P1 = missing required TLV Type	B101.0	66010100	CM: docsIf3CmEventNotif
Init (BPI+)	DOCSIS 1.0 CONFIG FILE	Alert	Invalid BP Configuration Setting Value: <P1> for Type: <P2><TAGS>	P1=The TLV Value for P2. P2 = The first Configuration TLV Type that contain invalid value.	B102.0	66010200	CM: docsIf3CmEventNotif
UCC							
UCC	UCC Request	Error	UCC-REQ received with invalid or out of range US channel ID<TAGS>		C01.0	67000100	
UCC	UCC Request	Error	UCC-REQ received unable to send UCC-RSP<TAGS>		C02.0	67000200	
DHCP, TOD and TFTP							
Init	TFTP	Critical	TFTP file complete- but doesn't enable 2.0 Mode - conflicts with current US channel type<TAGS>	For SYSLOG only: append: File name = <P1> P1 = file name of TFTP file	D11.0	68001100	

Annex D Extended Network Monitoring Requirements (Normative)

D.1 Overview

This annex addresses:

- Proactive Network Maintenance and Enhanced Signal Quality Monitoring requirements for plant conditions.
- Reporting of Service Flow Latency statistics.

D.1.1 PNM

PNM tests require downstream connectivity to provide accurate results. Therefore, if there are changes in the system during an active PNM test that modify the RCS of the CM or errors in the system that cause the CM to switch the primary downstream channel to a back-up primary downstream channel, the data collected by the PNM test would be invalidated. The CM MUST abort all active PNM tests if the CM receives a DBC-REQ message changing its RCS while one or more PNM tests are active. The CM MUST abort all active PNM tests if a loss of downstream connectivity occurs that causes the CM to switch its primary downstream channel to its backup primary downstream channel when one or more PNM tests are active.

D.1.2 Latency Reporting

The CM is capable of calculating the estimated queueing delay on a per service flow basis. The operator can enable statistics gathering and reporting for these latency estimates for individual service flows. These statistics are reported in the form of a configurable histogram.

D.2 Proactive Network Maintenance Information Model

This section defines the Proactive Network Maintenance objects including the associated attributes.

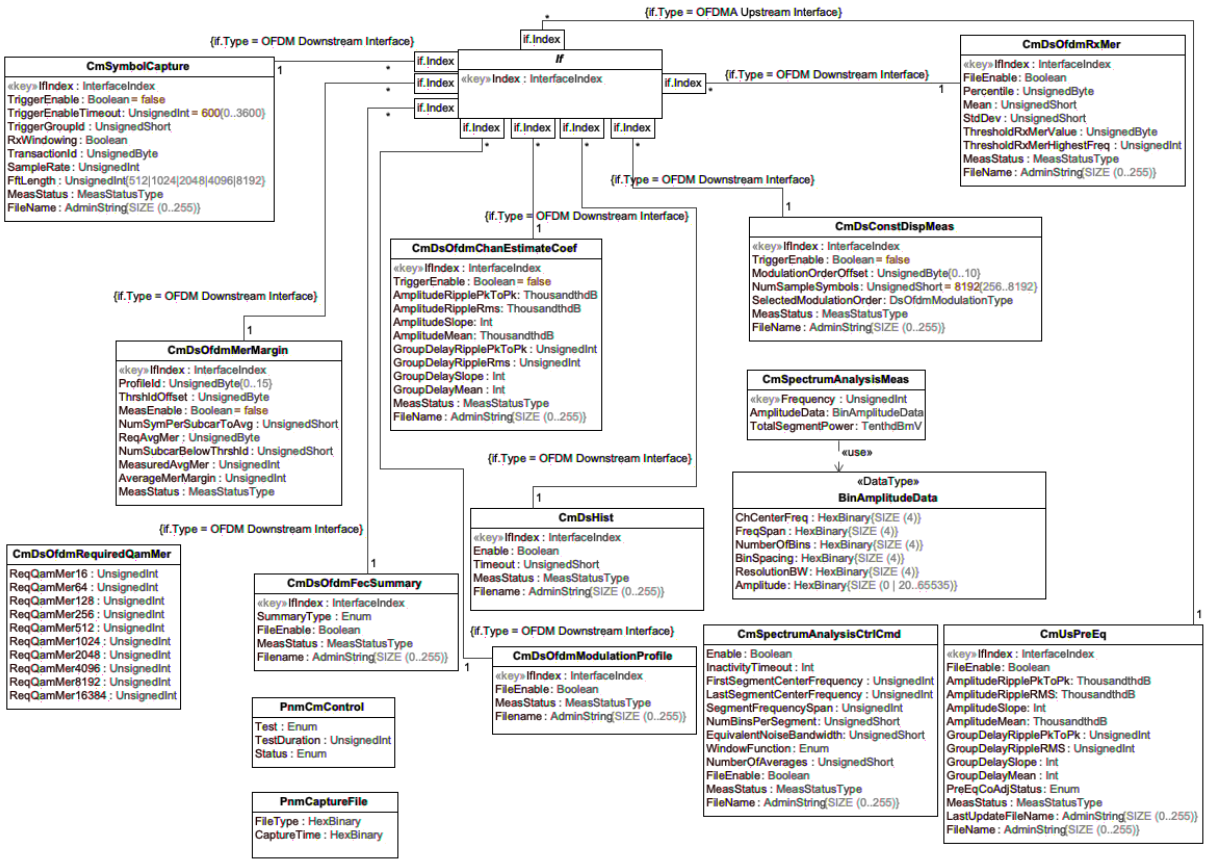


Figure 9 - Proactive Network Maintenance Information Model

D.2.1 Type Definitions

This section defines data types used in the object definitions for the Proactive Network Maintenance information model, as well as in the PNM data files generated by the CM.

Table 87 - Data Type Definitions

Data Type Name	Base Type	Permitted Values
BinAmplitudeData	HexBinary	SIZE(0 | 20..65535)
ComplexData	HexBinary	
RxMerData	HexBinary	
BinAmplitudeFileData	HexBinary	
MeasStatusType	Enum	other(1) inactive(2 busy(3) sampleReady(4) error(5) resourceUnavailable(6) sampleTruncated(7) interface Modification(8)
FecSummaryData	HexBinary	
ModulationProfileData	HexBinary	

D.2.1.1 BinAmplitudeData

This data type represents a sequence of spectral amplitudes. Each spectral amplitude value corresponds to a bin.

The format of the bin measurement is as follows.

Sequence of:

4 bytes: ChCenterFreq

The center frequency of the upstream channel.

4 bytes: FreqSpan

The width in Hz of the band across which the spectral amplitudes characterizing the channel are measured.

4 bytes: NumberOfBins

The number of data points or bins that compose the spectral data. The leftmost bin corresponds to the lower band edge, the rightmost bin corresponds to the upper band edge, and the middle bin center is aligned with the center frequency of the analysis span.

4 bytes: BinSpacing

The frequency separation between adjacent bin centers. It is derived from the frequency span and the number of bins or data points. The bin spacing is computed from

$$BinSpacing = \frac{FrequencySpan}{NumberOfBins - 1}$$

The larger the number of bins the finer the resolution.

4 bytes: ResolutionBW

The resolution bandwidth or equivalent noise bandwidth of each bin. If spectral windowing is used (based on vendor implementation), the bin spacing and resolution bandwidth would not generally be the same.

n bytes: Amplitude (2 bytes * NumberOfBins)

A sequence of two-byte elements corresponding to the bin amplitude. Each element represents the spectral amplitude in relation to a fixed reference segment power of 0 dBmV.

Each bin element amplitude value format is 2's complement which provides a range of -327.68 dB to 327.67 dB amplitude value for the bin measurement.

D.2.1.2 ComplexData

This data type uses 16-bit fixed-point, fractional, twos-complement notation to represent each of the I (real) and Q (imaginary) components of a complex number. When viewed as a 32-bit number in a file, the I component is the most significant 16 bits and the Q component is the least significant 16 bits. Positive or negative input values exceeding the number format are clipped on I and Q independently (no rollover).

The fixed-point format uses "sm.n" notation to indicate the location of the binary point in the I and Q components. Reading from left to right there is a sign bit, m integer bits, the assumed location of the binary point, and n fractional bits. If the I or Q components have less than 16 available bits of input data, the binary point of the input data is adjusted to match the sm.n specification in the 16-bit field, with the MSBs sign-filled if required, and the LSBs zero-filled if required.

Examples of sm.n notation: With s1.14 format, the numerical value "1" corresponds to hex pattern 0x4000. With s2.13 format, the numerical value "1" corresponds to 0x2000. With s3.12 format, the numerical value "1" corresponds to 0x1000.

Example of a complex number with s2.13 format on I and Q components: 0x2400F800 represents the complex number 1.125 - 0.25 j, where "j" is the square root of -1; that is, I = 1.125 and Q = -0.25.

When this data type represents a sequence of complex numbers values for an OFDM channel, it is expressed as a series of complex numbers for each subcarrier (active or excluded) from the lowest-frequency active subcarrier to the highest-frequency active subcarrier with no gaps.

D.2.1.3 RxMerData

This data type represents a sequence of received modulation error ratio (RxMER) values for a downstream OFDM channel at the CM. The data is expressed as a series of RxMerDataValues - one RxMerDataValue for each subcarrier (active or excluded) from the lowest-frequency active subcarrier to the highest-frequency active subcarrier with no gaps.

D.2.1.3.1 RxMerDataValue

This data type is used to express RxMER and is defined below:

- A single byte value with units of QuarterDb (e.g., a value of 23.75 dB = 0x5F)
- Range 0 to 63.5 dB in ¼ dB steps
- The value 0xFF is used to indicate no measurement is available for a given subcarrier
- Any value over 63.5 dB is reported as 63.5 dB
- Any value below 0 dB is reported as 0 dB

D.2.1.4 BinAmplitudeFileData

This data type is used to represent magnitude of the Spectrum Analysis bins. The values are expressed as 16-bit two's complement values in units of hundredthsDb. The dB values are referenced to 0 dBmV.

D.2.1.5 MeasStatusType

This data type is used to determine the state of a measurement. The MeasStatusType values are interpreted as follows:

- other - indicates any state not described below.
- inactive - indicates that a test is not currently in progress.
- busy - indicates that a test has been started and is in progress.
- sampleReady - indicates that a test has completed and that the measurement data is ready.
- error - indicates that there was an error starting or during the test and any test data, if available, may not be valid.
- resourceUnavailable - indicates that the test could not be started due to lack of CM test platform resources.
- sampleTruncated - indicates that the size of the requested data exceeded file size supported.
- interfaceModification - indicates that the interface numbering is changed due to DBC message or when Primary backup is changed.

D.2.1.6 FecSummaryData

This data type is used in the CmDsOfdmFecSummaryForProfile Bulk Data file. Multiple instances of this data type are included in the file as indicated by the NumberOfProfiles Header Element for the file. The data type uses unsigned integers encoded in hexadecimal notation as follows:

D.2.1.6.1 Profile Id - 1 byte

The Profile Id corresponding to this set of FEC data.

D.2.1.6.2 NumberOfCodewordSets - 2 bytes

The number of sets of CodewordEntries for this profile; either 600 or 1440 depending on the summary type. Each set of entries consists of a value for the TotalCodewords, CorrectedCodewords and UncorrectableCodewords for the particular one second- or one-minute interval.

D.2.1.6.3 Codeword Entries:

- CodewordSetTimeStamp - 4 bytes

  The time when this codeword set was collected. This field provides value if there is a signal interruption or if the profile is changed while the SummaryData is being collected. The value is expressed in epoch time (also known as 'unix time') and is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

- TotalCodewords - 4 bytes

  The total number of codewords recorded on this profile during the 1-second or 1-minute interval.

- CorrectedCodewords - 4 bytes

  The number of corrected codewords recorded on this profile during the 1-second- or 1-minute interval.

- UncorrectableCodewords - 4 bytes

  The number of uncorrectable codewords recorded on this profile during the 1-second or 1-minute interval.

D.2.1.7 ModulationProfileData

This data type represents a sequence of Subcarrier Assignments for a downstream OFDM channel at the CM from subcarrier zero up to and including the highest excluded subcarrier, corresponding to the subcarrier index equal to the number of FFT samples-1, with no gaps.

Subcarrier Assignment	Modulation Order Type
0	zero bit-loaded
1	Continuous Pilot
2	QPSK
3	reserved
4	16-QAM
5	reserved
6	64-QAM
7	128-QAM
8	256-QAM
9	512-QAM
10	1024-QAM
11	2048-QAM
12	4096-QAM
13	8192-QAM
14	16384-QAM
16	exclusion
20	PLC

D.2.1.7.1 Profile Id - 1 byte

The Profile Id corresponding to this set of Modulation Profile Data.

D.2.1.7.2 Length - 2 bytes

Total length in bytes of the modulation schemes.

D.2.1.7.3 Subcarrier Assignment Scheme Type - 1 byte

This byte represents the type of subcarrier assignment scheme type.

Table 88 - Modulation Scheme Types

Modulation Scheme Type	Value
0	Subcarrier Range Modulation
1	Subcarrier Skip Modulation
2-255	Reserved

D.2.1.7.3.1 Subcarrier Assignment Range

This scheme represents a range of contiguous subcarriers of the same subcarrier assignment type.

D.2.1.7.3.1.1 Subcarrier Assignment - 1 byte

This value represents the subcarrier assignment of the listed number of subcarriers.

D.2.1.7.3.1.2 Subcarrier Assignment Index Length - 2 bytes

The number of contiguous subcarriers of the same modulation type.

Table 89 - Subcarrier Range Modulation Scheme

Element	Size
Subcarrier Assignment Scheme Type	1 byte (Value 0x00)
Subcarrier Assignment	1 byte
Number of Subcarriers	2 bytes

D.2.1.7.3.2 Subcarrier Skip Modulation Scheme

This attribute indicates whether the skip modulation method is used. If true, the modulation order of the subcarriers in the range is alternating between the Main Modulation and Skip Modulation.

D.2.1.7.3.2.1 Main Modulation Order - 1 byte

This attribute indicates the main modulation order 2x. The main modulation is the modulation order of the first, the third, the fifth, etc., subcarriers in the range.

D.2.1.7.3.2.2 Skip Modulation Order - 1 byte

This attribute indicates the modulation order 2x for every other subcarrier in the range.

D.2.1.7.3.2.3 Skip Modulation Number of Subcarriers - 2 bytes

The total number of subcarriers including both the Main Modulation and Skip Modulation.

Table 90 - Subcarrier Skip Modulation Scheme

Element	Size
Subcarrier Assignment Scheme Type	1 byte (Value 0x01)
Main Modulation Order	1 byte
Skip Modulation Order	1 byte
Skip Modulation Number of Subcarriers	2 bytes

D.2.2 PnmCaptureFile

PNM tests usually generate files to report measurements or test results. The results file includes header information that is common to all types of PNM tests and fields, and data that are specific to the type of PNM test. The abstract PnmCaptureFile object defines the attributes and format of the header information common to all PNM test files described below. The File header fields are right-justified within the field and left-padded with zero values if necessary.

Determination of maximum size of the PNM Capture File is left to vendor implementation.

Table 91 shows the format of the PNM capture file header, applied to any data file created by a PNM test.

Table 91 - PnmCaptureFile Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
FileType	HexBinary	R/O	SIZE (4)	N/A	N/A
MajorVersion	UnsignedByte	R/O	SIZE (1)	N/A	N/A
MinorVersion	UnsignedByte	R/O	SIZE (1)	N/A	N/A
CaptureTime	UnsignedInt	R/O	SIZE (4)	N/A	N/A

D.2.2.1 FileType

A four-byte hexadecimal identifier specific to the type of PNM test that generated the data file. Each FileType is registered in [CANN]. Filetypes prefixed by PNM (i.e., 504E4D) do not contain the MajorVersion and MinorVersion header elements. Filetypes prefixed with PNN (i.e., 504E4E) contain the MajorVersion and MinorVersion header elements.

D.2.2.2 MajorVersion

This attribute represents the file header version. This value is incremented by one when the header format is modified by this specification.

D.2.2.3 MinorVersion

This attribute is reserved for vendor-specific and vendor-defined version information.

D.2.2.4 CaptureTime

A four-byte hexadecimal field representing the epoch time (also known as 'unix time') which is the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

D.2.3 PnmCmControl

PNM tests have the potential to stress the limited resources of a CM. Hence, certain PNM tests should not be allowed to run in parallel. This set of attributes serves to serialize PNM tests within the CM. No data from these control objects is required to persist across device resets.

This mechanism provides the name of the current or last test run and an estimate of the duration of the test. It also provides an attribute to signal that a test is currently in progress and to limit other tests from starting.

Other PNM tests will work with these objects to control which test has access to the PNM resources.

Table 92 - PnmCmControl Object Attributes

Attribute Name	Type		Access	Type Constraints	Units	Default
PnmCmCtlTest	Enum		R/O	other(1), dsSpectrumAnalyzer(2) dsOfdmSymbolCapture(3), dsOfdmChanEstCoef(4), dsConstellationDisp(5), dsOfdmRxMERPerSubCar(6), dsOfdmCodewordErrorRate(7), dsHistogram(8), usPreEqualizerCoef(9)	N/A	N/A
PnmCmCtlTestDuration	UnsignedInt		R/O	N/A	seconds	N/A
PnmCmCtlStatus	Enum		R/O	other(1), ready(2), testInProgress(3), tempReject(4)	N/A	N/A

D.2.3.1 PnmCmCtlTest

This attribute represents the current PNM test. The value could represent the current test in-progress, or if no test is running, the last test that was attempted.

For any test that is not specifically covered by the enumeration for this object, but manipulates the 'PnmCmCtlStatus' object, the CM MUST set this object to a value of 'other' and update the 'PnmCmCtlTestDuration' object. The value of this object is not required to persist across device resets. After a reset and before any test is run, the object will return a value of 'other'.

Possible values for this object are listed below:

other(1) - Any PNM test not covered by other defined values

dsSpectrumAnalyzer(2) - Downstream Spectrum Analysis described in Annex D.2.4, CM Spectrum Analysis Objects

dsOfdmSymbolCapture(3) - Downstream OFDM Symbol Capture described in Annex D.2.5, CmSymbolCapture

dsOfdmChanEstCoef(4) - Downstream OFDM Channel Estimate Coefficient described in Annex D.2.6, CmDsOfdmChanEstimateCoef

dsConstellationDisp(5) - Downstream Constellation Display described in Annex D.2.7, CmDsConstDispMeas

dsOfdmRxMERPerSubCar(6) - Downstream OFDM Receive Modulation Error Ratio per Subcarrier described in Annex D.2.8, CmDsOfdmRxMer

dsOfdmCodewordErrorRate(7) - Downstream OFDM Codeword Error Rate described in Annex D.2.10, CmDsOfdmFecSummary

dsHistogram(8) - Downstream Histogram described in Annex D.2.12, CmDsHist

usPreEqualizerCoef(9) - Upstream Pre-equalizer Coefficients described in Annex D.2.13, CmUsPreEq

D.2.3.2 PnmCmCtlTestDuration

This attribute represents the number of seconds that the test specified in 'PnmCmCtlTest' spent with a 'PnmCmCtlStatus' of 'testInProgress'. This serves to provide a rough (seconds resolution) estimate of the time spent under test. If this object is read while the value of 'PnmCmCtlStatus' is 'testInProgress', then the CM MUST return the number of seconds since the test started in this object. This value is informative only and is not a guarantee of future performance.

D.2.3.3 PnmCmCtlStatus

This attribute represents the overall status of the PNM test platform.

Individual tests within the PNM test suite have their own specific objects to start and stop. For each test, defined by the data-type enumeration for 'PnmCmCtlTest', the CM MUST first check the status of this object before starting. If this object is set to any value other than 'ready', the CM MUST NOT start the test. If the CM was not able to start the test and the test has a 'MeasStatusType' object, the CM MUST set 'MeasStatusType' to a value of 'resourceUnavailable'.

If the test is allowed to start, the CM MUST change the value of this object to 'testInProgress'. When the test is no longer in progress, the CM MUST change the value of this object to a value other than 'testInProgress'.

This mechanism serves to allow only one instance of only one test to run at a time. There may be some tests that are not included by this control object. That is, if any test is not defined by the data-type enumeration for 'PnmCmCtlTest', then it is not included (unless otherwise stated by the test definition) and is not required to check or to change this object. If a test is not included or covered by this set of objects, then it is allowed to run regardless of any other tests in progress.

The possible values for this object are:

'other' - Any condition not otherwise defined

'ready' - The PNM platform is capable and ready to support a test; a test can be initiated

'testInProgress' - A PNM test is currently in progress

'tempReject' - A temporary condition exists that prohibits a test from starting, e.g., DBC

D.2.4 CM Spectrum Analysis Objects

This group of objects provides a CM downstream spectrum analysis function. Each measurement is a data collection event that provides the energy content of the signal at each frequency within a specified range. The result of a measurement is a table consisting of one or more rows. Each row corresponds to a capture of spectral data across a specified segment bandwidth. The frequency range of each segment is divided into bins, which are a discrete set of evenly spaced frequencies across the band. The width of each bin (resolution bandwidth) is generally equal to or slightly greater than the spacing between bins. Each bin has an associated amplitude value in the table, which represents the amount of energy measured in that frequency bin. The segments are constrained to be contiguous; that is, the start frequency of each segment equals the end frequency of the previous segment plus the bin spacing. Thus, the concatenation of all segments results in a wideband spectral analysis. The measurement table is updated at a rate that is vendor-specific. The measurement generally occurs prior to the point at which the received signal is demodulated. The measurement spectrum may or may not include the effects of receiver processing such as gain control, RF filtering, and matched filtering.

The CM MUST implement the CmSpectrumAnalysisCtrlCmd object.

The CM MUST implement the CmSpectrumAnalysisMeas object.

D.2.4.1 CmSpectrumAnalysisCtrlCmd

This object is used to configure the frequency spectral analysis in the CM.

Table 93 - CmSpectrumAnalysisCtrlCmd Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Enable	Boolean	R/W			false
InactivityTimeout	Int	R/W	0..86400	seconds	300
FirstSegmentCenterFrequency	UnsignedInt	R/W		Hz	93000000
LastSegmentCenterFrequency	UnsignedInt	R/W		Hz	993000000
SegmentFrequencySpan	UnsignedInt	R/W	1000000..900000000	Hz	7500000
NumBinsPerSegment	UnsignedShort	R/W	2..2048	bins-per-segment	256
EquivalentNoiseBandwidth	UnsignedShort	R/W	50..500	Hundredths of bin spacing	150

Attribute Name	Type	Access	Type Constraints	Units	Default
WindowFunction	Enum	R/W	other(0), hann(1), blackmanHarris(2), rectangular(3), hamming(4), flatTop(5), gaussian(6), chebyshev(7)		
NumberOfAverages	UnsignedShort	R/W	1..1000		1
FileEnable	Boolean	R/W			false
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE(1..255)		

D.2.4.1.1 Enable

This attribute is used to enable or disable the spectrum analyzer feature. Setting this attribute to true triggers the CM to initiate measurements for the spectrum analyzer feature based on the other configuration attributes for the feature. By default, the feature is disabled unless explicitly enabled. Note that the feature may be disabled by the system under certain circumstances if the spectrum analyzer would affect critical services. In such a case, the attribute will return 'false' when read, and will reject sets to 'true' with an error. Once the feature is enabled, any configuration operations (e.g., write operations to configuration objects) might not be effective until the feature is re-enabled.

D.2.4.1.2 InactivityTimeout

This attribute controls the length of time after the last spectrum analysis measurement before the feature is automatically disabled. If set to a value of 0, the feature will remain enabled until it is explicitly disabled.

D.2.4.1.3 FirstSegmentCenterFrequency

This attribute controls the center frequency of the first segment for the spectrum analysis measurement.

The frequency bins for this segment lie symmetrically to the left and right of this center frequency. If the number of bins in a segment is odd, the segment center frequency lies directly on the center bin. If the number of bins in a segment is even, the segment center frequency lies halfway between two bins.

Changing the value of this object may result in changes to the CmSpectrumAnalysisMeas object, as described in the description field for the object.

Note that if this object is set to an invalid value, the device may return an error of inconsistentValue, or may adjust the value of the object to the closest valid value.

D.2.4.1.4 LastSegmentCenterFrequency

This attribute controls the center frequency of the last segment of the spectrum analysis measurement.

The frequency bins for this segment lie symmetrically to the left and right of this center frequency. If the number of bins in a segment is odd, the segment center frequency lies directly on the center bin. If the number of bins in a segment is even, the segment center frequency lies halfway between two bins.

The value of the LastSegmentCenterFrequency is typically equal to the FirstSegmentCenterFrequency plus an integer number of segment spans as determined by the SegmentFrequencySpan.

Changing the value of this object may result in changes to the CmSpectrumAnalysisMeas object, as described in the description field for the object.

Note that if this attribute is set to an invalid value, the device may return an error of inconsistentValue, or may adjust the value of the attribute to the closest valid value.

D.2.4.1.5 SegmentFrequencySpan

This attribute controls the frequency span of each segment (instance) of the CmSpectrumAnalysisMeas object. If set to a value of 0, then a default span will be chosen based on the hardware capabilities of the device. Segments are contiguous from the FirstSegmentCenterFrequency to the LastSegmentCenterFrequency and the center frequency for each successive segment is incremented by the SegmentFrequencySpan. The number of segments is (LastSegmentCenterFrequency - FirstSegmentCenterFrequency)/SegmentFrequencySpan + 1. A segment is equivalent to an instance in the CmSpectrumAnalysisMeas object. The chosen SegmentFrequencySpan affects the number of instances in the CmSpectrumAnalysisMeas object. A more granular SegmentFrequencySpan may adversely affect the amount of time needed to query the instances in addition to possibly increasing the acquisition time.

Changing the value of this object may result in changes to the CmSpectrumAnalysisMeas object, as described in the description field for the object

Note that if this attribute is set to an invalid value, the device may return an error of inconsistentValue, or may adjust the value of the attribute to the closest valid value.

D.2.4.1.6 NumBinsPerSegment

This attribute controls the number of bins collected by the measurement performed for each segment (instance) of the CmSpectrumAnalysisMeas object.

Note that if this attribute is set to an invalid value, the device may return an error of inconsistentValue, or may adjust the value of the attribute to the closest valid value.

D.2.4.1.7 EquivalentNoiseBandwidth

This attribute allows the user to request an equivalent noise bandwidth for the resolution bandwidth filter used in the spectrum analysis. This corresponds to the spectral width of the window function used when performing a discrete Fourier transform for the analysis.

The window function which corresponds to a value written to this attribute may be obtained by reading the value of the WindowFunction attribute.

If an unsupported value is requested, the device may return an error of inconsistentValue, or choose the closest valid value to the one which is requested. If the closest value is chosen, then a subsequent read of this attribute will return the actual value that is in use.

D.2.4.1.8 WindowFunction

This attribute controls or indicates the windowing function that will be used when performing the discrete Fourier transform for the analysis. The WindowFunction and the EquivalentNoiseBandwidth are related. If a particular WindowFunction is selected, then the EquivalentNoiseBandwidth for the function in use will be reported by the EquivalentNoiseBandwidth attribute. Alternatively, if an EquivalentNoiseBandwidth value is chosen and a WindowFunction function representing that EquivalentNoiseBandwidth is defined in the CM, that value will be reported in the WindowFunction object, or a value of 'other' will be reported. Use of "modern" windowing functions not yet defined will likely be reported as 'other'.

Note that all window functions may not be supported by all devices. If an attempt is made to set the attribute to an unsupported window function, or if writing of the WindowFunction object is not supported by an implementation, an error will be returned.

D.2.4.1.9 NumberOfAverages

This attribute controls the number of averages that will be performed on spectral bins. The average will be computed using the "leaky integrator" method, where reported bin value = alpha*accumulated bin values + (1-alpha)*current bin value. Alpha is one minus the reciprocal of the number of averages. For example, if N=25, then alpha = 0.96. A value of 1 indicates no averaging. Re-writing the number of averages will restart the averaging process. If there are no accumulated values, the accumulators are made equal to the first measured bin amplitudes.

The number of averages will be set by writing NumberOfAverages attribute. If an attempt is made to set the attribute to an unsupported number of averages, an error of inconsistentValue will be returned.

D.2.4.1.10 FileEnable

This attribute, when set to 'true', causes the CM to begin a Spectrum Analysis measurement with the parameters defined by the CmSpectrumAnalysisCtrlCmd set of attributes. In order to set FileEnable to true, the Enable needs to already be set to true.

When the measurement is completed successfully, a file is generated and is made available for transfer and the MeasStatus attribute is set to 'sampleReady'. The file will contain one complete snapshot of the spectrum data.

Setting this object to a value of 'false' instructs the CM to stop the measurement.

D.2.4.1.11 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.4.1.12 Filename

This attribute is the name of the file, at the CM and containing the spectrum analysis data, which is to be downloaded by the PNM server.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

If the value of this object is the DEFVAL (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the format would be:

PNMSpecAnData_<CM MAC address>_<epoch>

For example: PNMSpecAnData_0010181A2D11_1403405123

The data file is composed of a header plus the Spectrum Analysis Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 94 - CM Spectrum Analysis File Format

Element	Size
File type (value = 504E4D09)	4 bytes
Major Version (value = 1)	1 byte
Minor Version	1 byte
Capture Time	4 bytes
Channel ID	1 byte
CM MAC Address	6 bytes
FirstSegmentCenterFrequency	4 bytes
LastSegmentCenterFrequency	4 bytes
SegmentFrequencySpan	4 bytes
NumBinsPerSegment	2 bytes

Element	Size
EquivalentNoiseBandWidth	2 bytes
WindowFunction	2 bytes
Length (in bytes) of SpectrumAnalysis Data	4 bytes
SpectrumAnalysisData	BinAmplitudeFileData

If the size of the data the CM is commanded to collect would exceed the maximum file-size limit of the data collection mechanism, the CM will limit the file size accordingly and set the 'MeasStatus' attribute to a value of 'sampleTruncated'. If this occurs, the file will contain valid data, from the beginning of the capture, but will represent fewer bins than configured for the measurement.

D.2.4.1.13 File Header Element Definitions

D.2.4.1.13.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.4.1.13.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.4.1.13.3 Capture Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

D.2.4.1.13.4 Channel ID

Channel ID is not valid for this file type; needs to be set to 0.

D.2.4.1.13.5 CM MAC Address

The CM MAC address.

D.2.4.1.13.6 FirstSegmentCenterFrequency

This element is a copy of the FirstSegmentCenterFrequency attribute.

D.2.4.1.13.7 LastSegmentCenterFrequency

This element is a copy of the LastSegmentCenterFrequency attribute.

D.2.4.1.13.8 SegmentFrequencySpan

This element is a copy of the SegmentFrequencySpan attribute.

D.2.4.1.13.9 NumBinsPerSegment

This element is a copy of the NumBinsPerSegment attribute.

D.2.4.1.13.10 EquivalentNoiseBandWidth

This element is a copy of the EquivalentNoiseBandWidth attribute.

D.2.4.1.13.11 WindowFunction

This element is a copy of the attribute.

D.2.4.1.13.12 Length (in bytes) of SpectrumAnalysis Data

This element indicates the size of the data which follows.

D.2.4.2 CmSpectrumAnalysisMeas

This object provides a list of the spectral amplitude measurements taken across the requested range of center frequencies. The table represents a full scan of the spectrum with each row corresponding to a spectral capture of one segment of the spectrum.

Table 95 - CmSpectrumAnalysisMeas Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Frequency	UnsignedInt	key		Hz	
AmplitudeData	BinAmplitudeData	R/O			
TotalSegmentPower	TenthdBmV	R/O		dBmV	

D.2.4.2.1 Frequency

This key indicates the center frequency of the spectral analysis segment which is represented by this instance.

D.2.4.2.2 AmplitudeData

This attribute provides a list of the spectral amplitudes as measured at the center frequency specified by the Frequency attribute.

The frequency bins are ordered from lowest to highest frequencies covering the frequency span. Information about the center frequency, frequency span, number of bins and resolution bandwidth are included to provide context to the measurement point.

Bin Amplitudes are reported in units of 0.01dB.

D.2.4.2.3 TotalSegmentPower

This attribute provides the total RF power present in the segment with the center frequency equal to the Frequency index and the span equal to the SegmentFrequencySpan. The value represents the sum of the spectrum power in all of the associated bins. The value is computed by summing power (not dB) values and converting the final sum to TenthdBmV.

D.2.5 CmSymbolCapture

The purpose of downstream symbol capture is to provide partial functionality of a network analyzer to analyze the response of the cable plant from the CM's perspective.

At the CM, the received I and Q time-domain samples of one full OFDM symbol before the FFT, not including the guard interval, are captured, and made available for analysis. This capture will result in a number of data points equal to the FFT length in use, time aligned for receiver FFT processing. The number of captured samples can be reduced for narrower channels if the sampling rate, which is implementation dependent, is reduced. The capture includes a bit indicating if receiver windowing effects are present in the data. The time domain samples are expressed as 16-bit two's complement numbers using s3.12 format. The CM samples are scaled such that the average power of the samples is approximately 1, in order to avoid excessive clipping and quantization noise.

Capturing the input and output of the cable plant is equivalent to a wideband sweep of the channel, which permits full characterization of the linear and nonlinear response of the downstream plant. The MAC provides signaling via the PLC Trigger Message to ensure that the same symbol is captured at the CMTS and CM.

The Downstream Symbol Capture is controlled by setting the 'TriggerEnable' attribute. The status of the capture is obtained by reading the value of the 'MeasStatus' attribute.

This table will have a row for each ifIndex for the modem.

Table 96 - CmSymbolCapture Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
ifIndex	InterfaceIndex	Key			
TriggerEnable	Boolean	R/W			False
TriggerEnableTimeout	UnsignedInt	R/W	0..3600	seconds	600
TriggerGroupId	UnsignedShort	R/W			0
RxWindowing	Boolean	R/O			
TransactionId	UnsignedByte	R/O			
SampleRate	UnsignedInt	R/O		Hz	
FftLength	UnsignedInt	R/O	512 | 1024 | 2048 | 4096 | 8192		
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE (1..255)		empty string

D.2.5.1 ifIndex

This attribute is the interface index of the OFDM downstream channel and is a key to provide an index into the table.

D.2.5.2 TriggerEnable

If this attribute is set to a value of 'true', the CM MUST begin looking for the presence of the Trigger Message Block in the PLC with a Group ID matching the CM's TriggerGroupId. The TriggerEnable is a one-shot enable and the attribute is internally disabled when a PLC containing a Group ID matching the CM's TriggerGroupId in a Trigger Message Block is received.

If this attribute is set to a value of 'false', the CM MUST stop looking for the presence of the Trigger Message Block in the PLC with a Group ID matching the CM's TriggerGroupId. In this case, the value of the MeasStatus attribute will be set to 'inactive'.

When read, the CM MUST return a value of 'true' if the CM is actively looking for the presence of the Trigger Message Block in the PLC with a Group ID matching the CM's TriggerGroupId. Otherwise, the CM MUST return 'false'.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, the CM MUST return 'inconsistentValue' if this attribute is set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

This attribute returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

Default values for FileName, TriggerEnableTimeout, and TriggerGroupId attributes are defined; thus, this attribute may be set to 'true' without explicitly setting these values. Care should be taken to ensure these values are correct for the desired test case.

Setting this attribute to a value of 'true' will change the value of the MeasStatus attribute to 'busy'.

D.2.5.3 TriggerEnableTimeout

This attribute is used to disable the TriggerEnable if no PLC containing a Group ID matching the CM's TriggerGroupId in a Trigger Message Block is received within the timeout period. The CM MUST timeout the test after TriggerEnableTimeout seconds from the time when the TriggerEnable was set to 'true'. A value of 0 indicates that no timeout is enforced and the CM MUST enable the test until it completes, which in this case means the trigger could be enabled indefinitely. This value can only be changed while a test is not in progress. An attempt to set this value while the value of 'MeasStatus' is 'busy' will return 'inconsistentValue'.

D.2.5.4 TriggerGroupId

This attribute is used to associate a CM with a group of CMs expected to perform Symbol Capture measurements for the designated symbol.

This value can only be changed while a test is not in progress. If the CM receives an attempt to set this value while the value of MeasStatus is 'busy', it MUST return 'inconsistentValue' for this attribute.

D.2.5.5 RxWindowing

This attribute is a flag indicating if vendor proprietary receiver windowing was enabled during the capture.

D.2.5.6 TransactionId

This attribute is the Transaction ID sent by the CMTS in the Trigger Message Block. The CMTS increments this field by one on each trigger message that is sent, rolling over at value 255. Prior to completion of a measurement this attribute has no meaning.

D.2.5.7 SampleRate

This attribute is the FFT sample rate in use by the CM for the channel. Typically, the sample rate for the downstream channel will be 204.8 MHz.

D.2.5.8 FftLength

This attribute is the FFT length in use by the CM for the channel. Typically, this value is 4096 or 8192 for the Downstream Channel.

D.2.5.9 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.5.10 Filename

This attribute is the name of the file, at the CM and containing captured symbol data, which is to be downloaded by the PNM server.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

If the value of this object is the DEFVAL (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the format would be:

```
PNMSymCap_<CM MAC address>_<epoch>
```

For example: PNMSymCap_0010181A2D11_1403405123

The data file is composed of a header plus the Symbol Capture Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 97 - CM Symbol Capture File Format

Element	Size
File type (value = 504E4E01)	4 bytes
Capture Time	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
DS Channel Id	1 byte
CM MAC	6 bytes
Subcarrier zero Frequency in Hz	4 bytes
SampleRate in Hz	4 bytes
FFT Size	4 bytes
TriggerGroupId	2 bytes
Transaction ID	1 byte
Length (in bytes) of Capture Data	4 bytes
Capture Data	ComplexData

D.2.5.10.1 File Header Element Definitions

D.2.5.10.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.5.10.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.5.10.1.3 Capture Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

D.2.5.10.1.4 DS Channel Id

This element represents the channel Id of the downstream channel for which the symbol capture data apply.

D.2.5.10.1.5 CM MAC Address

The CM MAC address.

D.2.5.10.1.6 Subcarrier zero Frequency in Hz

The center frequency of subcarrier zero of the OFDM channel.

D.2.5.10.1.7 SampleRate in Hz

This element is a copy of the SampleRate attribute.

D.2.5.10.1.8 FFT Size

This element is a copy of the FftLength attribute.

D.2.5.10.1.9 TriggerGroup ID

This element is a copy of the TriggerGroupId attribute.

D.2.5.10.1.10 Transaction ID

This element is a copy of the TransactionId attribute.

D.2.5.10.1.11 Length (in bytes) of Captured Data

This element indicates the size of the complexData which follows.

D.2.5.10.1.12 Capture Data

This element refers to the complexData values of the CM Symbol Capture. The data is expressed in s3.12 fixed point notation. The CM should adjust the scaling such that the average power of the samples is approximately 1, in order to avoid excessive clipping and quantization noise.

D.2.6 CmDsOfdmChanEstimateCoef

The purpose of this table is for the CM to report its estimate of the downstream channel response. The reciprocals of the channel response coefficients are typically used by the CM as its frequency-domain downstream equalizer coefficients. The channel estimate consists of a single complex value per subcarrier. The channel response coefficients are expressed as 16-bit two's complement numbers using 2.13 format. The CM samples are scaled such that the average power of the samples is approximately 1, in order to avoid excessive clipping and quantization noise.

Summary metrics (slope, ripple, and mean) are defined in order to avoid having to send all coefficients on every query. The summary metrics are calculated when the corresponding MIB is queried. A Coefficient filename and trigger are provided to obtain the channel coefficients.

This table will have a row for each ifIndex for the modem.

Table 98 - CmDsOfdmChanEstimateCoef Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
TriggerEnable	Boolean	R/W			False
AmplitudeRipplePkToPk	ThousandthdB	R/O		dB	
AmplitudeRippleRms	ThousandthdB	R/O		dB	
AmplitudeSlope	Int	R/O		ThousandthdB/MHz	
AmplitudeMean	ThousandthdB	R/O		dB	
GroupDelayRipplePkToPk	UnsignedInt	R/O		0.001 nsec	
GroupDelayRippleRms	UnsignedInt	R/O		0.001 nsec	
GroupDelaySlope	Int	R/O		0.001 nsec/MHz	
GroupDelayMean	Int	R/O		0.001 nsec	
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE (0..255)		empty string

D.2.6.1 IfIndex

This attribute is the interface index of the OFDM downstream channel interface and is a key to provide an index into the table.

D.2.6.2 TriggerEnable

Setting this object to a value of 'true' instructs the CM to begin collection and storing the channel estimate coefficients into the file specified by the FileName object.

Setting this object to a value of 'false' instructs the CM to stop storing channel estimate coefficients in the file. When read, this object returns 'true' if the CM is actively storing channel estimate coefficients in the file. Otherwise, it returns 'false'.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, this object returns 'inconsistentValue' if set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

This attribute returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

A default value for FileName is defined; thus, this object may be set to 'true' without explicitly setting that value.

Setting this attribute to a value of 'true' will change the value of MeasStatus to 'busy'.

D.2.6.3 AmplitudeRipplePkToPk

This attribute represents the value of the peak to peak ripple in the magnitude of the equalizer coefficients [PHYv4.0]. The slope component calculated for the AmplitudeSlope is subtracted from the frequency domain data and the peak to peak Ripple is calculated from the resultant data. This attribute represents the ripple across the entire OFDM channel.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.6.4 AmplitudeRippleRms

This attribute represents the value of the RMS ripple in the magnitude of the equalizer coefficients. The slope component calculated for the AmplitudeSlope is subtracted from the frequency domain data and the RMS ripple is calculated from the resultant data. This attribute represents the ripple across the entire OFDM channel.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.6.5 AmplitudeSlope

This attribute represents the slope in 0.001 dB per MHz in the magnitude of the equalizer coefficients. The slope is calculated as the slope of a linear least squares fit of the frequency domain data. This attribute represents the slope across the entire OFDM channel.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.6.6 AmplitudeMean

This attribute represents the Mean, in 0.001 dB, of the magnitude of the equalizer coefficients.

D.2.6.7 GroupDelayRipplePkToPk

This attribute represents the peak to peak Group Delay Ripple expressed in units of 0.001 nsec. This attribute represents the group delay variation across the entire OFDM channel. The slope component calculated for the GroupDelaySlope is subtracted from the frequency domain data and the peak-to-peak ripple is calculated from the resulting data.

D.2.6.8 GroupDelayRippleRMS

This attribute represents the RMS value of the Group Delay Ripple expressed in units of 0.001 nsec. This attribute represents the group delay variation across the entire OFDM channel. The slope component calculated for the GroupDelaySlope is subtracted from the frequency domain data and the RMS ripple is calculated from the resulting data. This attribute is not stored in the data file.

D.2.6.9 GroupDelaySlope

This attribute represents the slope in 0.001 nsec per MHz in the group delay of the equalizer coefficients. This attribute represents the slope across the entire OFDM channel.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.6.10 GroupDelayMean

This attribute represents the mean of the group delay, in units of 0.001 nsec.

D.2.6.11 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.6.12 FileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. The data is stored as 16-bit integers for the I and Q data.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970. Hence, the format would be:

PNMChEstCoef_<CM MAC address>_<epoch>

For example: PNMChEstCoef_0010181A2D11_1403405123

The data file is composed of a header plus the Chan Estimate Coefficient Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 99 - Channel Estimate Coefficient File Format

Element	Size
File type (value = 504E4E02)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
Capture Time	4 bytes
DS Channel Id	1 byte
CM MAC Address	6 bytes
Subcarrier zero frequency in Hz	4 bytes
FirstActiveSubcarrierIndex	2 bytes
Subcarrier spacing in kHz	1 byte
Length in bytes of coefficient data	4 bytes
Chan estimate coefficient data	Complex Data

D.2.6.12.1 File Header Element Definitions

D.2.6.12.1.1 Capture Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

D.2.6.12.1.2 DS Channel Id

This element represents the channel Id of the downstream channel for which the Channel Estimate data apply.

D.2.6.12.1.3 CM MAC Address

The CM MAC Address.

D.2.6.12.1.4 Subcarrier zero Frequency in Hz

This element is the center frequency of subcarrier zero of the OFDM channel.

D.2.6.12.1.5 Subcarrier spacing in kHz

This element is the subcarrier spacing for the OFDM channel - 25 or 50 kHz.

D.2.6.12.1.6 FirstActiveSubcarrierIndex

This element is the subcarrier index of the lowest subcarrier in the Encompassed Spectrum of the channel.

D.2.6.12.1.7 Length in bytes of coefficient data

This element indicates the size of the complexData which follows.

D.2.6.12.1.8 Chan estimate coefficient data

This element refers to the complexData values of the CmOfdmChanEstimateCoef object. These data elements typically represent the reciprocal of CM DS frequency-domain equalizer coefficients with one complex value for each subcarrier within the Encompassed Spectrum. The data is expressed in s2.13 fixed point notation. The CM should adjust the scaling such that the average power of the samples is approximately 1, in order to avoid excessive clipping and quantization noise. If there are excluded subcarriers within the Encompassed Spectrum, the CM MUST report a value of 0x8000 for both of the I and Q values for those excluded subcarriers. If the CM calculates 0x8000 for both of the I and Q values for an active subcarrier, then the CM MUST report a value of 0x8001 for both of the I and Q values for that subcarrier.

D.2.6.12.1.9 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.6.12.1.10 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.7 CmDsConstDispMeas

The downstream constellation display provides received QAM constellation points for display. Equalized soft decisions (I and Q) at the slicer input are collected over time, possibly subsampling to reduce complexity, and made available for analysis. This measurement is intended for data subcarriers only. Up to 8192 samples are provided for each query; additional queries can be made to further fill in the plot.

The complex Constellation Display values are expressed as 16-bit two's complement numbers using s2.13 format. The CM samples are scaled such that the average power of the QAM constellation is approximately 1, in order to avoid excessive clipping and quantization noise.

The object controls the CM capturing and reporting received soft-decision samples, for a single selected constellation from the set of profiles it is receiving, within a single OFDM downstream channel.

This table will have a row for each ifIndex for the modem.

Table 100 - CmDsConstDispMeas Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
TriggerEnable	Boolean	R/W			False
ModulationOrderOffset	UnsignedByte	R/W	0..10		0
NumSampleSymbols	UnsignedShort	R/W			8192
SelectedModulationOrder	DsOfdmModulationType	R/O			
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE (1..255)		empty string

D.2.7.1 IfIndex

This attribute is the interface index of the downstream channel and is a key to provide an index into the table.

D.2.7.2 TriggerEnable

Setting this attribute to a value of 'true' instructs the CM to begin collection and storing the constellation points into the file specified by the FileName attribute.

Setting this attribute to a value of 'false' instructs the CM to stop storing constellation points in the file.

When read, this attribute returns 'true' if the CM is actively storing constellation points in the file. Otherwise, it returns 'false'.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, this object returns 'inconsistentValue' if set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

This attribute returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

A default value for FileName is defined; thus, this attribute may be set to 'true' without explicitly setting the FileName value.

Setting this attribute to a value of 'true' will change the value of the MeasStatus to 'busy'.

D.2.7.3 ModulationOrderOffset

This attribute specifies an offset from the lowest order modulation for the data subcarriers in any of the profiles in the downstream channel. If the lowest order modulation order that the CM was receiving was 1024-QAM and the ModulationOrderOffset was zero, then the CM would capture the soft decision samples for all of the subcarriers which were using 1024-QAM modulation order. If the ModulationOrderOffset was 1, then the CM would capture the soft decision samples for all of the subcarriers using the next highest modulation order in use for the profiles in the downstream channel.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

D.2.7.4 NumSampleSymbols

This attribute tells the CM how many soft decision samples of OFDM subcarriers with the specified modulation order are captured.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

D.2.7.5 SelectedModulationOrder

This read-only attribute provides the actual Modulation Order that will be used for the Constellation display based on the selected ModulationOrderOffset.

D.2.7.6 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.7.7 FileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. The data is stored as 16-bit integers for the I and Q data.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

Hence, the format would be:

PNMDSConDisp_<CM MAC address>_<epoch>

For example: PNMDSConDisp_0010181A2D11_1403405123

The data file is composed of a header plus the Constellation Display Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 101 - Constellation Display File Format

Element	Size
File type (value = 504E4E03)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
Capture Time	4 bytes
DS Channel Id	1 byte
CM MAC Address	6 bytes
Subcarrier zero frequency in Hz	4 bytes
Actual modulation order*	2 bytes
Number of sample symbols	2 bytes
Subcarrier spacing in kHz	1 byte
Length in bytes of display data	4 bytes
Constellation display data	Complex Data

*DsOfdmModulationType

D.2.7.7.1 File Header Element Definitions

D.2.7.7.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.7.7.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.7.7.1.3 Capture Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

D.2.7.7.1.4 DS Channel Id

This element represents the channel Id of the downstream channel for which the constellation data were collected.

D.2.7.7.1.5 CM MAC Address

The CM MAC Address.

D.2.7.7.1.6 Subcarrier zero Frequency in Hz

This element is the center frequency of subcarrier zero of the OFDM channel.

D.2.7.7.1.7 Actual Modulation Order

This element is a copy of the SelectedModulationOrder attribute.

D.2.7.7.1.8 Number of sample symbols

This element is a copy of the NumSampleSymbols attribute.

D.2.7.7.1.9 Subcarrier spacing in kHz

This element is the OFDM subcarrier spacing.

D.2.7.7.1.10 Length in bytes of display data

This element is the number of bytes of the ComplexData.

D.2.7.7.1.11 Constellation Display Data

This element refers to the complexData values of the Constellation Display data. The data is expressed in s2.13 fixed point notation. The CM should adjust the scaling such that the average power of the QAM constellation (not including pilots), assuming equally probable QAM symbols, is approximately 1, in order to avoid excessive clipping and quantization noise.

D.2.8 CmDsOfdmRxMer

This object provides measurements of the receive modulation error ratio (RxMER) for each subcarrier.

Each subcarrier RxMER value consists of one byte, which represents the RxMER value with range 0 to 63.5 dB in 0.25 dB steps. If some subcarriers (such as exclusion bands) cannot be measured by the CM, the CM indicates that condition by reporting a value of 0xFF for the RxMER for those subcarriers. Any measured RxMER value below 0 dB is clipped to 0 dB (0x00), and any measured value above 63.5 dB is clipped to 63.5 dB (0xFE).

The CM has the capability of generating a file containing all subcarrier RxMER values. The CM also reports four summary metrics (Mean, StdDev, ThresholdRxMerValue and ThresholdRxMerHighestFreq) which can be used to determine whether to request the file of all subcarrier RxMER values.

Mathematical notation for the calculation of RxMER is provided in [PHYv4.0]: "Downstream Receive Modulation Error Ratio (RxMER) Per Subcarrier".

This object will have an instance for each ifIndex for the modem.

The operator may choose two methods of operation for this object. The first is obtaining detailed Rx MER statistics on a per-subcarrier basis. To do this, the operator sets a filename for the data file in the FileName attribute and then sets the FileEnable attribute to true. The Operator may also choose to perform an ad-hoc measurement of the summary metrics for this channel. In this case, the operator performs a query of the object or performs individual queries on one or more of the following attributes:

- Mean
- StdDev
- ThresholdRxMerValue
- ThresholdRxMerHighestFreq

When the CM receives a query request on the Mean, StdDev, ThresholdRxMerValue, or ThresholdRxMerHighestFreq attributes, the CM MUST perform a measurement for the attribute being requested. When the CM receives a query on the CmDsOfdmRxMer object, the CM MUST perform a measurement for all of these attributes. The CM MAY update the Mean, StdDev, ThresholdRxMerValue, and ThresholdRxMerHighestFreq attributes when a query is performed on any of these attributes.

Table 102 - CmDsOfdmRxMer Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
FileEnable	Boolean	R/W			False
Percentile	UnsignedByte	R/W		percent	2
Mean	HundredthdB	R/O	0..65535	dB	
StdDev	HundredthdB	R/O	0..65535	hundredthdB	
ThresholdRxMerValue	UnsignedByte	R/O		quarterDb	
ThresholdRxMerHighestFreq	UnsignedInt	R/O		Hz	
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE (1..255)		empty string

D.2.8.1 IfIndex

This attribute is the interface index of the downstream channel and is a key to provide an index into the table.

D.2.8.2 FileEnable

This attribute causes the CM to begin the RxMer measurement for the purpose of creating a file of RxMer data. When the measurement is complete the FileEnable attribute is set internally to false by the CM.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, this object returns 'inconsistentValue' if set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

Setting this value to 'true' will change the value of the MeasStatus to 'busy' while the test is in progress.

This attribute returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

A default value for FileName is defined; thus, this object may be set to 'true' without explicitly setting the FileName value.

Setting this attribute to a value of 'false' instructs the CM to stop the measurement.

This attribute returns 'true' if the CM is actively taking a measurement; otherwise it returns 'false'.

D.2.8.3 Percentile

This attribute specifies the percentile (such as 2nd percentile or 5th percentile) of all active subcarriers in an OFDM channel at which the ThresholdRxMerValue occurs. That is, (Percentile) % of the subcarriers have RxMER <= ThresholdRxMerValue.

D.2.8.4 Mean

This attribute is the mean of the dB values of the RxMER measurements of all active subcarriers. The mean is computed directly on the dB values as follows:

Mean = sum of (RxMER dB values) / number of RxMER values

D.2.8.5 StdDev

This attribute is the standard deviation of the dB values of the RxMER measurements of all active subcarriers. The standard deviation is computed directly on the dB values as follows:

StdDev = sqrt(sum of (RxMER dB values - RxMER_mean)^2 / number of RxMER values)

D.2.8.6 ThresholdRxMerValue

This attribute is the RxMER value corresponding to the specified Percentile value. The CM sorts the subcarriers in ascending order of RxMER, resulting in a post-sorting subcarrier index ranging from 1 to the number of active

subcarriers. If the percentile value corresponds to a non-integer post-sorting subcarrier index, the post-sorting index is truncated (floor function is applied); that is, the post-sorting index is selected which is the greatest integer less than or equal to the corresponding percentile value. For example, if there are 3677 active subcarriers and the 2nd percentile is specified, the CM computes floor(3677*0.02) = 73. That is, the RxMER value of the 73rd subcarrier in the sorted list is associated with the 2nd percentile.

D.2.8.7 ThresholdRxMerHighestFreq

This attribute is the frequency in Hz of the highest-frequency subcarrier having RxMER = ThresholdRxMerValue.

D.2.8.8 Example Calculations of ThresholdRxMerValue and ThresholdRxMerHighestFreq

As a first example, assume there are 3800 active subcarriers in the OFDM channel being measured, and the Percentile attribute is set to 2%. Using a sorting process, the CM finds the ThresholdRxMerValue corresponding to the specified Percentile. Assume for this example that ThresholdRxMerValue = 25 dB, that is, 2% of the subcarriers have RxMER <= 25 dB. The CM also finds the highest frequency subcarrier having RxMER = 25 dB. Assume for this example that this subcarrier has frequency 702 MHz. The CM reports ThresholdRxMerValue = 25 dB and ThresholdRxMerHighestFreq = 702 MHz.

As a second more detailed example, consider a simplified case of an OFDM channel with 50 kHz subcarrier spacing and only 32 active subcarriers; in reality the number of subcarriers in a channel will be in the hundreds or thousands. To further simplify the example, 1 dB resolution is assumed for RxMER; in reality the RxMER values will have 0.25 dB resolution. Assume the sequence of subcarriers is sorted in ascending order of RxMER as in Table 103.

Table 103 - Example of RxMER Summary Statistics with 32 Subcarriers

Subcarrier Index (Post Sorting)	RxMER (dB)	Subcarrier Frequency (MHz)
1	19	501.10
2	20	501.00
3	20	501.05
4	20	501.15
5	21	501.20
6	21	501.25
7	21	501.30
8	22	501.35
9	23	500.95
10	23	501.40
11	23	501.55
12	24	500.90
13	24	501.45
14	25	500.75
15	25	500.80
16	25	500.85
17	25	501.50
18	28	500.70
19	29	500.65
20	30	500.00
21	30	500.50
22	30	500.60
23	31	500.45
24	31	500.55

Subcarrier Index (Post Sorting)	RxMER (dB)	Subcarrier Frequency (MHz)
25	32	500.05
26	33	500.15
27	33	500.40
28	34	500.10
29	34	500.35
30	35	500.25
31	35	500.30
32	36	500.20

For this example, let the specified Percentile = 20; that is, to find the subcarrier corresponding to the 20th percentile (in reality, typical values will be 2nd or 5th percentile). The CM computes floor(20% of 32 subcarriers) = 6, meaning that the RxMER value of the 6th subcarrier in the sorted list corresponds to the specified Percentile; this subcarrier has an RxMER value of 21 dB. To determine ThresholdRxMerHighestFreq, the CM finds the frequency of the highest-frequency subcarrier having RxMER = 21 dB. There are 3 subcarriers with RxMER = 21 dB, and of these, the one with the highest frequency is the 7th subcarrier in the sorted list, having frequency 501.30 MHz. The CM reports ThresholdRxMerValue = 21 dB and ThresholdRxMerHighestFreq = 501.30 MHz.

D.2.8.9 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.8.10 FileName

This attribute is the name of the file at the CM which is to be downloaded by the PNM server.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

If the value of this attribute is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the format would be:

PNMDsMer_<CM MAC address>_<epoch>

For example: PNMDsMer_0010181A2D11_1403405123

The data file is composed of a header plus the Subcarrier RxMER Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 104 - RxMER File Format

Element	Size
File type (value = 504E4E04)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte

Element	Size
Capture Time	4 bytes
DS Channel Id	1 byte
CM MAC Address	6 bytes
Subcarrier zero frequency in Hz	4 bytes
FirstActiveSubcarrierIndex	2 bytes
Subcarrier spacing in kHz	1 byte
Length in bytes of RxMER data	4 bytes
Subcarrier RxMER data	RxMerData

D.2.8.10.1 File Header Element Definitions

D.2.8.10.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.8.10.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.8.10.1.3 Capture Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

D.2.8.10.1.4 DS Channel Id

This element represents the channel Id of the downstream channel for which the RxMer data apply.

D.2.8.10.1.5 CM MAC Address

The CM MAC Address.

D.2.8.10.1.6 Subcarrier zero Frequency in Hz

This element is the center frequency of subcarrier zero of the OFDM channel.

D.2.8.10.1.7 FirstActiveSubcarrierIndex

This element is the subcarrier index of the lowest subcarrier in the Encompassed Spectrum of the channel.

D.2.8.10.1.8 Subcarrier spacing in kHz

This element is the OFDM subcarrier spacing.

D.2.8.10.1.9 Length in bytes of RxMER data

This element is the number of bytes of the RxMerData.

D.2.9 CmDsOfdmMerMargin

The purpose of this item is to provide an estimate of the MER margin available on the downstream data channel with respect to a modulation profile. The profile may be a profile that the modem has already been assigned or a candidate profile. This is similar to the MER Margin reported in the OPT-RSP Message [MULPIv4.0].
The CM calculates the Required Average MER for the profile based on the bit loading for the profile and the Required MER per Modulation Order provided in the CmDsOfdmRequiredQamMer Table. For profiles with mixed modulation orders, this value is computed as an arithmetic mean of the required MER values for each non-excluded subcarrier in the Modulated Spectrum. The CM then measures the RxMER per subcarrier and calculates the Average MER for the Active Subcarriers used in the Profile and stores the value as MeasuredAvgMer. The Operator may also compute the value for Required Average MER for the profile and set that value for the test.
The CM also counts the number of MER per Subcarrier values that are below the threshold determined by the

CmDsOfdmRequiredQamMer and the ThrshldOffset. The CM reports that value as NumSubcarriersBelowThrshld. This table will have a row for each ifIndex for the modem.

Table 105 - CmDsOfdmMerMargin Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
ProfileId	UnsignedByte	R/W	0..15		0
ThrshldOffset	UnsignedByte	R/W		quarterDb	0
MeasEnable	Boolean	R/W			False
NumSymPerSubcarToAvg	UnsignedShort	R/W			8
ReqAvgMer	UnsignedByte	R/W		quarterDb	0
NumSubcarBelowThrshld	UnsignedShort	R/O			
MeasuredAvgMer	UnsignedInt	R/O		hundredthDb	
AverageMerMargin	Int	R/O		hundredthDb	
MeasStatus	MeasStatusType	R/O			

D.2.9.1 IfIndex

This attribute is the interface index of the downstream channel and is a key to provide an index into the table.

D.2.9.2 DsProfileId

This attribute represents the Downstream Profile ID of the Profile. The profile may be a profile that the modem has already been assigned or a candidate profile. This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

D.2.9.3 ThreshldOffset

This attribute represents the number of dB below the CmDsOfdmRequiredQamMer value for a given modulation order that is likely to cause uncorrectable errors. Measurements of Subcarrier MER that are this number of dB or more below the CmDsOfdmRequiredQamMer for the Subcarrier for the profile being tested, will cause the CM to increment the count of the NumOfSubcarBelowThreshld attribute.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

D.2.9.4 MeasEnable

This attribute causes the CM to begin measurement. When the measurement is complete, the MeasEnable attribute is set internally to 'false' by the CM.

This value is only allowed to be set to 'true' if the value of 'MeasStatus' is a value other than 'busy' for this row AND for any row in the table. That is, only one row in the table is allowed to be 'true' at the same time. Setting this value to 'true' will change the value of 'MeasStatus' to 'busy'.

Setting this object to a value of 'false' instructs the CM to stop the measurement.

This object returns 'true' if the CM is actively taking a measurement; otherwise it returns 'false'.

This object returns 'inconsistentValue' if set to 'true' while the value of 'MeasStatus' is a value of 'busy' for this row OR for any row in the table. That is, only one row in the table is allowed to be 'true' at the same time.

This object returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

Setting this object to a value of 'true' will change the value of the 'MeasStatus' to 'busy'.

D.2.9.5 NumSymbolsPerSubcarrierToAverage

This attribute represents the number of symbols that will be used in the calculation of the average MER per subcarrier. This value can only be changed while a test is not in progress. An attempt to set this value while the value of 'MeasStatus' is 'busy' will return 'inconsistentValue'.

D.2.9.6 ReqAvgMer

This attribute represents the minimum required average MER. This value can either be computed by the entity requesting the test or be computed by the test when the attribute uses the default value 0. The CM subtracts this value from the MeasuredAvgMer to obtain the AverageMerMargin. If this value is not provided by the PNM server (i.e., it is the default value of zero), the CM computes the value as the average of the required MER for all of the subcarriers, based on the values in the CmDsOfdmRequiredQamMer table, for the Modulation order used for each of the subcarriers for the profile being analyzed. For profiles with mixed modulation orders, this value is computed by averaging as an arithmetic mean of the required MER values for each set of non-excluded subcarriers in the modulated spectrum.

D.2.9.7 NumSubcarriersBelowThrshld

This attribute represents the count of the number of subcarriers which were below the ReqAvgMer for the ThreshldOffset attribute.

D.2.9.8 MeasuredAvgMer

This attribute is the arithmetic mean of all MER dB values measured over all of the subcarriers in the Modulated Spectrum. That is, the average is taken of all the dB values.

D.2.9.9 AverageMerMargin

This attribute represents the difference between the MeasuredAvgMer and the ReqAvgMer.

D.2.9.10 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the data is ready for evaluation.

D.2.9.11 Example

The CM calculates the Required Average MER based on the bit loading for the profile and the Required MER per Modulation Order provided in the CmDsOfdmRequiredQamMer Table. The CM then measures the RxMER per subcarrier and calculates the Average MER for the Active Subcarriers used in the Profile and stores the value as MeasuredAvgMer.

Attributes used in this example:

MeasuredAvgMer - average of RxMER per Subcarrier values in dB

ReqAvgMer - Assigned or derived from profile modulation orders and CmDsOfdmRequiredQamMer Table

ThreshldOffset - number subtracted from RequiredQamMer to calculate NumSubcarBelowThrshld
AverageMerMargin - Result of subtracting ReqAvgMer from MeasuredAvgMer

NumSubcarriersBelowThrshld - Total number of subcarriers for which the RxMER Per Subcarrier is less than the RequiredQamMer value minus the ThreshldOffset

Assumptions: Candidate Profile has 3800 Active Subcarriers in the Modulated Spectrum. The bit loading of 1900 of the subcarriers is 4096-QAM with a Required MER of 41 dB. The remainder of the 1900 subcarriers are 2048-QAM with Required MER of 38 dB. The Required Average MER is calculated as (41 * 1900 + 38 * 1900)/3800 = 39.5 dB. If the MeasuredAvgMer Per Subcarrier = 42 dB, then the AverageMerMargin is 42 – 39.5 = 2.5 (250 hundredthsdB).

Further, if 10 of the 4096-QAM and 10 of the 2048-QAM subcarriers have a Measured RxMER value equal to 36 dB and the ThreshldOffset is set to 4 dB, then the CM will report a value of 10 for the NumSubcarriersBelowThrshld since the criteria for counting the NumSubcarriersBelowThrshld for 4096-QAM would be 41-4=37, while the criteria for counting the NumSubcarriersBelowThrshld for 2048-QAM would be 37-4 = 33 dB.

D.2.10 CmDsOfdmFecSummary

The purpose of this item is to provide a series of codeword error rate measurements on a per profile basis over a set period of time.

This table will have a row for each ifIndex for the modem.

Table 106 - CmDsOfdmFecSummary Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
SummaryType	Enum	R/W	other(1), interval10min(2), interval24hr(3)		interval10min(2)
FileEnable	Boolean	R/W			false
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE (1..255)		empty string

D.2.10.1 IfIndex

This attribute is the interface index of the downstream interface and is a key to provide an index into the table.

D.2.10.2 SummaryType

This attribute is the type of summary test to be performed. If set to interval10min(2), when enabled, the CM MUST return the codeword data samples recorded once per second over the most recent 10-minute period for a maximum of 600 samples. If set to interval24hr(3), when enabled, the CM MUST return the codeword data samples recorded once per minute over the most recent 24-hour period for a maximum of 1440 samples. The CM begins collecting samples in the background when it becomes operational. The CM returns the collected data when the test is enabled. If the CM has not been in the operational state for the entire SummaryType period, it will not have collected the full data set for the SummaryType period. In this case, the CM will return the data that it has available since becoming operational, which will be less than the maximum size permitted for the SummaryType period.

Data samples are cleared on CM reset or power cycle. The CM MUST maintain the FEC Summary Data through a Re-Init MAC. When the CM is maintaining FEC Summary data across a Re-Init MAC operation, there will be a gap in the CodewordSetTimeStamp values and the total time for interval10min and interval24hr, SummaryTypes could be greater than 10 minutes or 24 hours, respectively. It is possible that the DS Channel associated with the ifIndex for which FEC Summary data is being collected could change due to a Re-Init MAC event. If this unlikely event were to occur, the CM will report the DS Channel Id currently associated with the ifIndex, in the header of the file being sent to the PNM server.

Reference: Downstream FEC Statistics [PHYv4.0]

D.2.10.3 FileEnable

When this attribute is set to 'true', the CM MUST begin the SummaryType codeword error summary test. While the test is in progress, the CM MUST return a MeasStatus value of 'busy'. When the measurement is complete, the CM MUST set the FileEnable attribute to 'false'. If this attribute is set to 'false' during a test, the CM MUST stop the test.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, the CM MUST return 'inconsistentValue' if set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

This attribute returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

A default value for FileName is defined; thus, this object can be set to 'true' without explicitly setting the FileName value.

The CM MUST return the value of 'true' if it is actively taking a measurement; otherwise it MUST return 'false'.

D.2.10.4 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the data is ready for evaluation.

D.2.10.5 FileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. This value can only be changed while a test is not in progress. The CM MUST return 'inconsistentValue' when it receives an attempt to set this value while the value of MeasStatus is 'busy'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, the CM MUST store the test results in a file with a name consisting of the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the format would be:

PNMFecSum_<CM MAC address>_<epoch>

For example: PNMFecSum_0010181A2D11_1403405123

The data file is composed of a header plus the FEC Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 107 - Downstream FEC Summary File Format

Element	Size or Data Type
File type (value = 504E4E08)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
DS Channel Id	1 byte
CM MAC Address	6 bytes
SummaryType	1 byte
Number of Profiles	1 byte
FEC Data	FecSummaryData

D.2.10.5.1 File Header Element Definitions

D.2.10.5.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.10.5.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.10.5.1.3 DS Channel Id

This element represents the channel Id of the downstream channel for which these FEC statistics apply.

D.2.10.5.1.4 CM MAC Address

The CM MAC Address.

D.2.10.5.1.5 SummaryType

This element is a copy of the Summary Type attribute for the MIB.

D.2.10.5.1.6 NumberOfProfiles

This element is number of profiles in the CM for which Codeword summaries are provided. This number indicates how many instances of the FecSummaryData type are included in the file.

The CM MUST include FEC data arrays for every profile assigned by the CCAP.

D.2.10.5.1.7 FEC Data

The set of FEC data encoded using the FecSummaryData type. There is one set of data per Profile ID as indicated by the NumberOfProfiles.

Example:

This is an example showing the FECData encoding for two profiles. In this example, there are only 5 codeword sets for each profile, whereas normally there would be 600 or 1440 depending on the Summary Type. The four columns of data correspond to the values for the CodewordSetTimeStamp, TotalCodewords, CorrectedCodewords and UncorrectableCodewords, respectively. The values in the example are expressed as decimal integer values, but they are encoded using 32-bit hexadecimal notation in the file. The order of codewordSets in the file is from oldest to newest, meaning that the codewordSet following the Header Elements is the oldest set.

Profile Id = 255

Number of Codeword Sets = 5

1456252719	5311230	0	0
1456252720	5268830	0	0
1456252721	5226430	0	0
1456252722	5184030	0	0
1456252723	5140830	0	0

Profile Id = 0

Number of Codeword Sets = 5

1456252719	221896	0	0
1456252720	220128	0	0
1456252721	218353	0	0
1456252722	216587	0	0
1456252723	214777	0	0

Hex Notation:

ff 05 56 CC A7 2F 00 51 0A FE 00 00 00 00 00 00 00 00 56 CC A7 30 00 50 65 5E 00 00 00 00 00 00 00 00, etc.

D.2.11 CmDsOfdmRequiredQamMer

The purpose of this item is to provide a target MER value for each downstream OFDM modulation order to be used in determining the SNR Margin for the Candidate Downstream Profile. The QamMer attributes are expressed in units of a quarter dB.

Table 108 - CmDsOfdmRequiredQamMer Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
ReqMerQam16	UnsignedInt	R/W		quarterDb	60
ReqMerQam64	UnsignedInt	R/W		quarterDb	84

Attribute Name	Type	Access	Type Constraints	Units	Default Value
ReqMerQam128	UnsignedInt	R/W		quarterDb	96
ReqMerQam256	UnsignedInt	R/W		quarterDb	108
ReqMerQam512	UnsignedInt	R/W		quarterDb	122
ReqMerQam1024	UnsignedInt	R/W		quarterDb	136
ReqMerQam2048	UnsignedInt	R/W		quarterDb	148
ReqMerQam4096	UnsignedInt	R/W		quarterDb	164
ReqMerQam8192	UnsignedInt	R/W		quarterDb	184
ReqMerQam16384	UnsignedInt	R/W		quarterDb	208

D.2.11.1 ReqMerQam16

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.2 ReqMerQam64

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.3 ReqMerQam128

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.4 ReqMerQam256

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.5 ReqMerQam512

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.6 ReqMerQam1024

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.7 ReqMerQam2048

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.8 ReqMerQam4096

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.9 ReqMerQam8192

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.11.10 ReqMerQam16384

This attribute represents the minimum required MER value for this Modulation Order. It is used in determining the SNR Margin for the Candidate Downstream Profile.

D.2.12 CmDsHist

The purpose of the downstream histogram is to provide a measurement of nonlinear effects in the channel such as amplifier compression and laser clipping. For example, laser clipping causes one tail of the histogram to be truncated and replaced with a spike. The CM captures the histogram of time domain samples at the wideband front end of the receiver (full downstream band). The histogram is two-sided; that is, it encompasses values from far-negative to far-positive values of the samples. The histogram will have 255 or 256 equally spaced bins. These bins typically correspond to the 8 MSBs of the wideband analog-to-digital converter (ADC) for the case of 255 or 256 bins. The histogram dwell count, a 32-bit unsigned integer, is the number of samples observed while counting hits for a given bin, and may have the same value for all bins. The histogram hit count, a 32-bit unsigned integer, is the number of samples falling in a given bin. The CM reports the dwell count per bin and the hit count per bin. When enabled, the CM computes a histogram with a dwell of at least 10 million samples at each bin in 30 seconds or less. The CM continues accumulating histogram samples until it is restarted, disabled or times out. If the highest dwell count approaches its 32-bit overflow value, the CM stops counting and sets the MeasStatus attribute to 'sampleReady'. The CM reports the capture time of the histogram measurement in the Header Elements of the file for TFTP download using the 32-bit Universal Time Code.

This table will have a row for each docsCableMaclayer type ifIndex for the modem.

Table 109 - CmDsHist Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
Enable	Boolean	R/W			false
Timeout	UnsignedShort	R/W		Seconds	1800
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/O	SIZE (0..255)		empty string

D.2.12.1 IfIndex

This attribute is the interface index of the RF MAC Interface and is a key to provide an index into the table.

D.2.12.2 Enable

Setting this attribute to a value of 'true' instructs the CM to begin collection of histogram data and when enabled, the CM continues producing new data at its own rate.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, this object returns 'inconsistentValue' if set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

Setting this object to a value of 'false' instructs the CM to stop the collection of histogram data.

This object returns 'true' if the CM is actively collecting histogram data. Otherwise, it returns 'false'.

This object returns 'inconsistentValue' if set to 'true' while the value of MeasStatus is a value of 'busy' for this row or for any row in the table. That is, only one row in the table is allowed to be 'true' at the same time.

This object returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

Setting this object to a value of 'true' will change the value of the MeasStatus to 'busy'.

A restart may be accomplished by setting this attribute to 'false' and then back to 'true'.

D.2.12.3 Timeout

This attribute sets a seconds time-out timer for capturing histogram data. This attribute is used to automatically clear the Enable attribute when the timeout expires. If the value of this attribute is zero, the CM will collect data until the value is changed, stopped, or until any dwell counter approaches its 32-bit rollover value. If the Timeout attribute is re-written while Enable is 'true', the CM will restart the timeout timer with the new Timeout value and continue collecting data. When the timeout expires, the Enable attribute will be set to 'false' and the capture will stop. When this happens, the data collected up to this point will be saved in the file defined by the DataFileName and the value of MeasStatus set to 'sampleReady'.

Setting this value does not start a capture. Captures can only be started by setting the Enable attribute.

If this attribute is written while the Enable object is 'true', the timer is restarted. If this attribute is set to a value of 'zero', there is no timeout and the collection of data will continue indefinitely.

This attribute returns the value with which it was last set.

D.2.12.4 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.12.5 FileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. The data is stored as 32-bit integers for the hit and dwell count values.

This value can only be changed while a test is not in progress. An attempt to set this value while the value of MeasStatus is 'busy' will return 'inconsistentValue'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

Hence, the format would be:

PNMHist_<CM MAC address>_<epoch>

For example: PNMHist_0010181A2D11_1403405123

The data file is composed of a header plus the Histogram Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 110 - Downstream Histogram File Format

Element	Size
File type (value = 504E4E05)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
Capture Time	4 bytes
CM MAC Address	6 bytes
Symmetry	1 byte
Length (in bytes) of Dwell Count Values	4 bytes
DwellCount values	(1-4096) * 4 bytes

Element	Size
Length (in bytes) of Hit Count Values	4 bytes
HitCount values	(1-4096) * 4 bytes

D.2.12.5.1 File Header Element Definitions

D.2.12.5.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.12.5.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.12.5.1.3 Capture Time

The time (epoch time) that the file was written. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

D.2.12.5.1.4 CM MAC Address

The CM MAC Address.

D.2.12.5.1.5 Symmetry

This attribute is used to indicate whether 256 or 255 bins were used for the measurement.

Even Symmetry = 'false' (default): The histogram has even symmetry about the origin. There is no bin center lying directly at the origin; rather, two bin centers straddle the origin at 0.5. All bins with indices 0-255 contain valid hit-count data. The histogram bin centers are offset from the corresponding 8-bit two's complement integer values by 1/2, that is, bin center = two's complement value + 0.5.

Odd Symmetry = 'true': The histogram has odd symmetry about the origin. There is a bin center lying at the origin. The bin with index 0 is not used and returns the value 0. The bins with indices 1 to 255 contain valid hit-count data. The histogram bin centers are located on the corresponding 8-bit two's complement integer values.

The following table shows the defined histogram bin centers for the cases of even and odd symmetry.

Table 111 - Histogram Bin Centers

Bin Index	Bin Center Even Symmetry	Bin Center Odd Symmetry
0	-127.5	not used
1	-126.5	-127
2	-125.5	-126
...	...	...
127	-0.5	-1
128	0.5	0
129	1.5	1
...	...	...
253	125.5	125
254	126.5	126
255	127.5	127

D.2.12.5.1.6 Length in bytes of Dwell Count Values

This element represents the number of Dwell Count Values which follow in the file.

D.2.12.5.1.7 Dwell Count Values

This element represents the Dwell Counts for each bin for the "Current" capture. The value is a sequence of 4-byte values. If the dwell count for all bins is the same, then only a single value is reported. The value for each bin is reported as a 32-bit value.

D.2.12.5.1.8 Length in bytes of Hit Count Values

This element represents the number of Hit Count Values which follow in the file.

D.2.12.5.1.9 Hit Count Values

This element represents the Hit Counts for each bin for the "Current" capture. The value represents a sequence of 4-byte values. If odd symmetry is used, then there will be 255 bins. The value for each bin is reported as a 32-bit value.

D.2.13 CmUsPreEq

This object provides access to CM upstream pre-equalizer coefficients. The CM pre-equalizer coefficients and the CMTS upstream adaptive equalizer coefficient update values, when taken together describe the linear response of the upstream cable plant for a given CM. During the ranging process, the CMTS computes adaptive equalizer coefficients based on upstream probes; these coefficients describe the residual channel remaining after any pre-equalization. The CMTS sends these equalizer coefficients to the CM as a set of Transmit Equalization Adjust coefficients as part of the ranging process. The CM Pre-Equalizer coefficients are expressed as 16-bit two's complement numbers using s2.13 format. The power averaged over all coefficients approximately 1, in order to avoid excessive clipping and quantization noise.

The Pre-Equalizer coefficient update values sent to the CM by the CMTS in the RNG-RSP are expressed as 16-bit two's complement numbers using S1.14 format.

The CM provides the capability to report its upstream pre-equalizer coefficients (full set or summary) upon request. The CM also provides the capability to report the most recent set of Transmit Equalization Adjust or Set coefficients which were applied to produce the reported set of upstream pre-equalizer coefficients. The CM indicates the status of the most recent Transmit Equalization Adjust or Set coefficients sent to it by the CMTS. If the CM was able to apply the coefficients, it sets the status to success(2). If the CM was unable to fully apply the adjustments (for example, due to excess tilt or ripple in the channel), and it was necessary for the CM to clip the coefficients, it sets the status to clipped(3). If the CM modified the coefficients other than by simply clipping them, it sets the status to other(1). If for some reason the CM is unable to apply the adjustments at all, it sets the status to rejected(4).

The 'TriggerEnable' attribute is used to create files. Other attributes are updated as read.

This table will have a row for each ifIndex for the modem.

Table 112 - CmUsPreEq Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
FileEnable	Boolean	R/W			False
AmplitudeRipplePkToPk	ThousandthdB	R/O		dB	
AmplitudeRippleRMS	ThousandthdB	R/O		dB	
AmplitudeSlope	Int	R/O		ThousandthdB/MHz	
AmplitudeMean	ThousandthdB	R/O		dB	
GroupDelayRipplePkToPk	UnsignedInt	R/O		0.001 nsec	
GroupDelayRippleRms	UnsignedInt	R/O		0.001 nsec	
GroupDelaySlope	Int	R/O		0.001 nsec/MHz	
GroupDelayMean	Int	R/O		0.001 nsec	

Attribute Name	Type	Access	Type Constraints	Units	Default Value
PreEqCoAdjStatus	Enum	R/O	other(1) success(2) clipped(3) rejected(4)		
MeasStatus	MeasStatusType	R/O			
LastUpdateFileName	AdminString	R/W	SIZE (0..255)		empty string
FileName	AdminString	R/W	SIZE (0..255)		empty string

D.2.13.1 IfIndex

This attribute is the interface index of the upstream OFDMA interface and is a key to provide an index into the table.

D.2.13.2 FileEnable

This attribute causes the files defined by the 'Filename' and the 'LastUpdateFilename' attributes to be created. The files, once created, are available via the 'docsPnmCmBulkFileTable' mechanism. The 'MeasStatus' object can be checked to determine the status of this attribute.

This attribute is subject to the rules specified by the PnmCmCtlStatus attribute. Therefore, this object returns 'inconsistentValue' if set to 'true' while the value of PnmCmCtlStatus is any value other than 'ready'.

Setting this value to 'true' will change the value of the MeasStatus to 'busy' while the file generation is in progress.

This object returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

Default values for Filename and LastUpdateFilename are defined; thus, this object may be set to 'true' without explicitly setting these values.

This object returns 'true' if the CM is actively generating the files; otherwise, it returns 'false.'

SUMMARY METRICS:

The summary metrics described in D.2.13.3 through D.2.13.10 are not calculated unless a specific query of any of the summary metrics has been performed.

D.2.13.3 AmplitudeRipplePkToPk

This attribute represents the value of the peak to peak ripple in the magnitude of the equalizer coefficients. This attribute represents the ripple across the entire OFDMA channel. This value is not stored in the data files.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.13.4 AmplitudeRippleRMS

This attribute represents the value of the RMS ripple in the magnitude of the equalizer coefficients. This attribute represents the ripple across the entire OFDMA channel. This value is not stored in the data files.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.13.5 AmplitudeSlope

This attribute represents the slope in 0.001 dB per MHz in the magnitude of the equalizer coefficients. This attribute represents the slope across the entire OFDMA channel. This value is not stored in the data files.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.13.6 AmplitudeMean

This attribute represents the mean, in 0.001 dB, of the magnitude of the equalizer coefficients.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.13.7 GroupDelayRipplePkToPk

This attribute represents the peak to peak Group Delay Ripple expressed in units of 0.001 nsec. This attribute represents the group delay variation across the entire OFDMA channel. This value is not stored in the data files. The slope component calculated for the GroupDelaySlope is subtracted from the frequency domain data and the peak-to-peak ripple is calculated from the resulting data.

D.2.13.8 GroupDelayRippleRMS

This attribute represents the RMS value of the Group Delay Ripple expressed in units of 0.001 nsec. This attribute represents the group delay variation across the entire OFDMA channel. The slope component calculated for the GroupDelaySlope is subtracted from the frequency domain data and the RMS ripple is calculated from the resulting data. This attribute is not stored in the data files.

D.2.13.9 GroupDelaySlope

This attribute represents the slope in 0.001 nsec per MHz in the group delay of the equalizer coefficients. This attribute represents the slope across the entire OFDM channel.

Note: An algorithm for calculating the ripple and slope for these measurements is provided in Annex D.4.

D.2.13.10 GroupDelayMean

This attribute represents the mean of the group delay in units of 0.001 nsec.

D.2.13.11 PreEquCoAdjStatus

This attribute represents whether the last set of Pre-Equalization coefficient adjustments were fully applied or were only partially applied due to excessive ripple or tilt in the Pre-Equalization coefficient values.

D.2.13.12 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the file is ready for transfer.

D.2.13.13 LastUpdateFileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. The data represents the data sent to the CM by the CMTS in the last RNG-RSP that contained Pre-Equalization Adjust or Set values and are stored as 16-bit integers for the I and Q data.

This object cannot be changed while a file generation is in progress. It will return a value of 'inconsistentValue' if set while the value of MeasStatus is set to a value of 'busy'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the format would be:

PNMUsPreEqLastUpdate_<CM MAC address>_<epoch>

For example: PNMUsPreEqLastUpdate_0010181A2D11_1403405123

The data file is composed of a header plus the Pre-EQ Update Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 113 - Last PreEqualization Update File Format

Element	Size
File type (value = 504E4E07)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
Capture Time	4 bytes
Upstream Channel Id	1 byte
CM MAC Address	6 bytes
CMTS MAC Address	6 bytes
Subcarrier zero frequency in Hz	4 bytes
FirstActiveSubcarrierIndex	2 bytes
Subcarrier spacing in kHz	1 byte
Length in bytes of Pre-EQ data	4 bytes
Pre-EQ Coefficient Update data	ComplexData

D.2.13.13.1 File Header Element Definitions

D.2.13.13.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.13.13.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.13.13.1.3 Collection Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

D.2.13.13.1.4 Upstream Channel Id

This element represents the Upstream Channel Id for which the Upstream PreEqualization Coefficient adjustment values apply.

D.2.13.13.1.5 CM MAC Address

The CM MAC Address.

D.2.13.13.1.6 CMTS MAC Address

This element contains the MAC Address of the downstream channel for which the CM is receiving MAPs and UCDs.

D.2.13.13.1.7 Subcarrier zero Frequency in Hz

This element is the center frequency of subcarrier zero of the OFDM channel.

D.2.13.13.1.8 FirstActiveSubcarrierIndex

This element is the subcarrier index of the lowest subcarrier in the Encompassed Spectrum of the channel.

D.2.13.13.1.9 Subcarrier spacing in kHz

This element is the OFDM subcarrier spacing.

D.2.13.13.1.10 Length in bytes of PreEq data

This element is the number of bytes of the PreEqData.

D.2.13.13.1.11 Pre-Eq Coefficient Update Data

This element refers to the complexData values of the Pre-Equalization Coefficient sent to the CM by the CMTS in the last RNG-RSP message which contained Pre-EQ coefficients. The data is expressed in s1.14 fixed point notation.

D.2.13.14 FileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. The data represents the current set of the Pre-Equalization values for all of the subcarriers within the Encompassed Spectrum and are stored as 16-bit integers for the I and Q data.

All formats are 16 bits on each of I and Q. 'sm.n' means sign bit, m integer bits, and n fractional bits.

Examples:

With s1.14 format, the numerical value '1' corresponds to hex pattern 0x4000.

With s2.13 format, the numerical value '1' corresponds to hex pattern 0x2000.

With s3.12 format, the numerical value '1' corresponds to hex pattern 0x1000.

Positive or negative values exceeding the number format are clipped on I and Q independently (no rollover).

This object cannot be changed while a file generation is in progress. It will return a value of 'inconsistentValue' if set while the value of MeasStatus is set to a value of 'busy'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default filename value is used, it is generated as the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970. Hence, the format would be:

PNMUsPreEq_<CM MAC address>_<epoch>

For example: PNMUsPreEq_0010181A2D11_1403405123

The data file is composed of a header plus the Pre-EQ Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 114 - Upstream PreEqualization File Format

Element	Size
File type (value = 504E4E06)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
Collection Time	4 bytes
Upstream Channel Id	1 byte
CM MAC Address	6 bytes
CMTS MAC Address	6 bytes

Element	Size
Subcarrier zero frequency in Hz	4 bytes
FirstActiveSubcarrierIndex	2 bytes
Subcarrier spacing in kHz	1 byte
Length in bytes of Pre-EQ data	4 bytes
Pre-EQ Coefficient Data	ComplexData

D.2.13.14.1 File Header Element Definitions

D.2.13.14.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.13.14.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.13.14.1.3 Collection Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

D.2.13.14.1.4 Upstream Channel id

This element represents the Upstream Channel Id for which the Upstream PreEqualization Coefficients apply.

D.2.13.14.1.5 CM MAC Address

The CM MAC Address.

D.2.13.14.1.6 CMTS MAC Address

This element contains the MAC Address of the downstream channel for which the CM is receiving MAPs and UCDs.

D.2.13.14.1.7 Subcarrier zero Frequency in Hz

This element is the center frequency of subcarrier zero of the OFDM channel.

D.2.13.14.1.8 FirstActiveSubcarrierIndex

This element is the subcarrier index of the lowest subcarrier in the Encompassed Spectrum of the channel.

D.2.13.14.1.9 Subcarrier spacing in kHz

This element is the OFDM subcarrier spacing.

D.2.13.14.1.10 Length in bytes of PreEq data

This element is the number of bytes of the PreEqData.

D.2.13.14.1.11 Pre-Eq Coefficient Data

This element refers to the complexData values of the CM Pre-Equalization Coefficients. The data is expressed in s2.13 Fixed Point notation. Note: the power averaged over all coefficients is normalized to 1 [PHYv4.0].

D.2.14 CmDsOfdmModulationProfile

The purpose of this object is to provide the per subcarrier bit loading of the modulation profiles.

This table will have a row for each downstream interface ifIndex for the modem.

Table 115 - CmDsOfdmModulation Profile Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
FileEnable	Boolean	R/W			false
MeasStatus	MeasStatusType	R/O			
FileName	AdminString	R/W	SIZE (0..255)		empty string

D.2.14.1 IfIndex

This attribute is the index of the downstream interface and is a key to provide an index into the table.

D.2.14.2 FileEnable

When the FileEnable attribute is set to 'true', the CM MUST compile all the modulation profiles that are assigned by the CMTS. While the compilation is in progress, the CM MUST return a MeasStatus value of 'busy'. When the measurement is complete, the CM MUST set the FileEnable attribute to 'false'. If the FileEnable attribute is set to 'false' during a compilation, the CM MUST stop the compilation.

This attribute returns 'inconsistentValue' if set to 'true' while the CM is in DOCSIS Light Sleep (DLS) mode or the CM is in battery-backup mode.

A default value for FileName is defined; thus, this object can be set to 'true' without explicitly setting the FileName value.

The CM MUST return the FileEnable value of 'true' if it is actively compiling the modulation profile data. The CM MUST return the FileEnable value of 'false' if it is not actively compiling the modulation profile data.

D.2.14.3 MeasStatus

This attribute is used to determine the status of the measurement. The PNM server will query this value to determine when the data is ready for evaluation.

D.2.14.4 FileName

This attribute is the name of the file at the CM which is to be transferred to the PNM server. This value can only be changed while a test is not in progress. The CM MUST return 'inconsistentValue' when it receives an attempt to set the FileName value while the value of MeasStatus is 'busy'.

If the value of this object is the default value (empty string), then a default filename value will be used. Otherwise, the value set will be used as the filename. If a default filename is generated, then that value will be returned in this object and will represent the filename that was used for the test. All subsequent tests should set this object to a meaningful value or to an 'empty string' value (to generate a new default filename) before starting a new test.

If a default FileName value is used, the CM MUST store the test results in a file with a name consisting of the test name plus the CM MAC Address plus the 'epoch time'. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the format would be:

PNMModProfile_<CM MAC address>_<epoch>

For example: PNMModProfile_0010181A2D11_1403405123

The data file is composed of a header plus the Modulation Profile Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Table 116 - Downstream OFDM Modulation Profile File Data

Element	Size or Data Type
File type (value = 504E4E0A)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version (Value = 0)	1 byte
Capture Time	4 bytes
DS Channel Id	1 byte
CM MAC Address	6 bytes
Number of Profiles	1 byte
Subcarrier zero frequency in Hz	4 bytes
FirstActiveSubcarrierIndex	2 bytes
Subcarrier spacing in kHz	1 byte
Length in bytes of the Modulation Profile Data	4 bytes
Modulation Profile Data	ModulationProfileData

D.2.14.4.1 File Header Element Definitions

D.2.14.4.1.1 Major Version

The current file header version assigned the "value". The version is incremented by one when the file header format is modified by specification. The specification defined version value is 1.

D.2.14.4.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.2.14.4.1.3 Capture Time

The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC),Thursday, 1 January 1970.

D.2.14.4.1.4 DS Channel Id

This element represents the channel Id of the downstream channel for which the modulation profile data apply.

D.2.14.4.1.5 CM MAC Address

The CM MAC Address.

D.2.14.4.1.6 Number of Profiles

This element is number of profiles in the CM that indicates how many instances of the Modulation Profile Data type are included in the file.

The CM MUST count every Modulation Profile assigned by the CCAP, not including NCP.

D.2.14.4.1.7 Subcarrier Zero Frequency in Hz

This element is the center frequency of subcarrier zero of the OFDM channel.

D.2.14.4.1.8 FirstActiveSubcarrierIndex

This element is the subcarrier index of the lowest active subcarrier in the Occupied Bandwidth of the channel.

D.2.14.4.1.9 Subcarrier spacing in kHz

This element is the OFDM subcarrier spacing.

D.2.14.4.1.10 Length in bytes of Modulation Profile Data

This element is the number of bytes of the Modulation Profile Data.

D.2.14.4.1.11 Modulation Profile Data

This element represents the Modulation Profile Data defined in Section D.2.1.7.

D.3 Latency Reporting

This section defines the object models supporting Latency Reporting. CCAP and cable modem features and capabilities can be leveraged to enable measurement and reporting of latency estimates through each of the downstream service flows. With this information, operations personnel can monitor latency trends and adjust network configurations as appropriate.

This section defines the Latency statistics that provide histogram counts, maximum latencies, etc., for each enabled service flow. Statistics files can be enabled for TFTP /bulk file upload via the bulk data transfer mechanisms defined in Section D.5.

D.3.1 CmLatencyRpt

This object provides configuration of the CM upstream latency reporting. This enables the creation of latency report files, which consist of a series of latency measurements on a per Service Flow basis over a provisioned period of time.

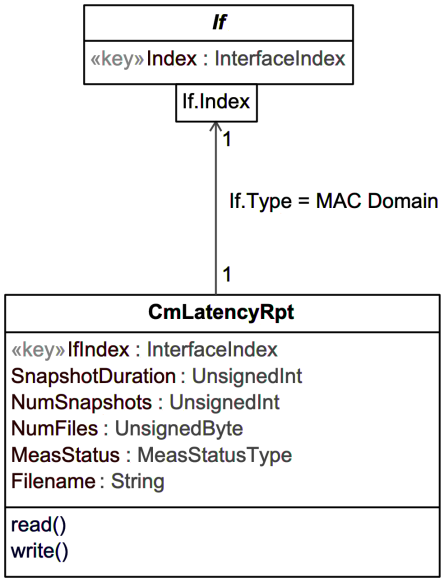


Figure 10 - Latency Report Information Model

The configuration attributes are described below.

Table 117 - CmLatencyRpt Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key	MAC Domain		
SnapshotDuration	UnsignedInt	R/W	1..3600	Seconds	60
NumSnapshots	UnsignedShort	R/W	1..2000		10
NumFiles	UnsignedByte	R/W	0..255	-	0
MeasStatus	MeasStatusType	R/O			
Filename	String	R/W	SIZE (0..231)		empty string

D.3.1.1 IfIndex

This attribute represents the interface index of the MAC Domain.

D.3.1.2 SnapshotDuration

This attribute represents the measurement duration of service flow latency estimates. One row of statistics per service flow is captured during this interval.

D.3.1.3 NumSnapshots

This attribute represents the number of Snapshots batched into a file. The total duration represented by the file is the product of the SnapshotDuration and NumSnapshots. Two examples follow.

- If SnapshotDuration is set to 1 second, and NumSnapshots is set to 10, when enabled, the CM returns the latency data samples recorded once per second over the next 10-sec period.
- If SnapshotDuration is set to 300 seconds (5 mins), and NumSnapshots is set to 288, when enabled, the CM returns the latency data samples recorded once per 5-minute interval over the next 24-hour period.

Certain combinations of SnapshotDuration and NumSnapshots could result in an excessive file generation rate or excessive file size. The CM MAY reject CmLatencyRpt attribute settings that cause an excessive processing load or resource requirement.

D.3.1.4 NumFiles

This attribute represents the number of file uploads performed as defined in Table 118 below.

Table 118 - NumFiles definition

NumFiles Attribute	
0	File generation disabled
1..254	Number of files to generate. Counts down from this value to zero
255	Unlimited file generation.

The CM MUST start generating a latency report file when this value is set to a non-zero value. The CM MUST close the current latency report file and decrement NumFiles by 1 when the SnapshotDuration * NumSnapshots period is complete. The CM MUST close the current latency report file and discontinue file generation if the NumFiles attribute is set to zero or decrements to zero.

D.3.1.5 MeasStatus

This attribute is used to determine the status of the measurement.

D.3.1.6 Filename

This attribute is the prefix for the name of the file that the CM creates to store latency histogram snapshots. This value can only be changed while a test is not in progress. The CM MUST return 'inconsistentValue' when it receives an attempt to set this value while the value of MeasStatus is 'busy'.

If the value of this object is the default value (empty string), then a default filename of 'UsLatencySum' will be used. Otherwise, the value set will be used as the filename prefix as described below.

The CM MUST store the test results in a file with a name consisting of the Filename plus the CM MAC Address plus the 'epoch time', using the format *<Filename>_<CM MAC address>_<epoch>*, with underscores between <Filename> and <CM MAC address> and <epoch> having significance. The epoch time (also known as 'unix time') is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Hence, the default format would be:

UsLatencySum_<CM MAC address>_<epoch>

For example: USLatencySum_0010181A2D11_1403405123

D.3.2 Latency Performance Reporting File Format

The data file is composed of a header plus the Latency Data. The header is composed of ordered fixed-length fields. Unless otherwise specified, the header fields contain hex values that are right-justified within the field. If necessary, the field is left-padded with zero values.

Syntax of the file is as follows:

Table 119 - Upstream Latency Summary File Format

Element	Size or Data Type
File type (value = 4C4C4401)	4 bytes
Major Version (Value = 1)	1 byte
Minor Version	1 byte
CM MAC Address	6 bytes
Number of LatencySummaryData objects (n)	1 byte
Latency Data	n*LatencySummaryData

D.3.2.1 File Header Element Definitions

D.3.2.1.1 Major Version

This element represents the current file header version. The current defined Major Version is 1. The version is incremented by one when the file header format is modified by specification.

D.3.2.1.2 Minor Version

The vendor-specified version information. The default value is zero if no vendor version is assigned.

D.3.2.1.3 Number of LatencySummaryData Objects

This number indicates how many instances of the LatencySummaryData type are included in the file. The CM MUST include an instance of LatencySummaryData for every service flow for which latency histogram calculation is turned on for any portion of the file reporting period. In addition, when the latency histogram calculation is disabled for a service flow during the file reporting period, the CM MUST conclude the current LatencySummaryData instance using the current time (as Snapshot End Timestamp) and histogram counts.

D.3.2.1.4 LatencyData

The set of Latency data encoded using the LatencySummaryData type described below. There is at least one set of data per histogram enabled Service Flow.

D.3.2.2 LatencySummaryData Type

This data type represents the estimated latency of packets for a service flow. The number of sanctioned packets and maximum measured latency during the measured interval are also reported.

Multiple instances of this data type are included in the file as indicated by the Number of LatencySummaryData objects header element for the file. In addition, if bin edges for a Service Flow are changed while Reporting is active, there will be multiple LatencySummaryData instances for the same Service Flow.

Table 120 - LatencySummaryData

Element	Size
SfId	4 bytes
SfLabel	16 bytes
Number of Bin Edges	1 byte

Element	Size
Array of Bin Edge Definitions	2bytes* Number of Bin Edges
NumSnapshots	2 bytes
First Snapshot Start Timestamp	4 bytes
Latency Snapshot Entry Format (1 entry per snapshot)	
Snapshot End Timestamp	4 bytes
Array of Bin counts	8bytes * (Number of Bin Edges+1)
MaxLatency	4 bytes
NumHistogramUpdates	4 bytes
SanctionedPkts	4 bytes
DroppedPkts	4 bytes
TotalEct0Pkts	4 bytes
Reserved	4 bytes
TotalEct1Pkts	4 bytes
CeMarkedEct1Pkts	4 bytes

D.3.2.2.1 SfId

The Service Flow identifier.

D.3.2.2.2 SfLabel

The Service Class Name signaled to the CM or the label configured by the operator by setting the SfLabel management object.

D.3.2.2.3 NumberOfBinEdges - 1 byte

The number of bin edges defined for the latency histogram for measurement on this Service Flow. The number of bin edges is one less than the number of bins used in the measurement, e.g., 9 bin edges will give 10 bins for measurement.

*D.3.2.2.4 Array of Bin Edge Definitions - 2 bytes * NumberOfBinEdges*

The array of bin edges defined for this latency histogram measurement on this Service Flow. This element is expressed as an array of bin edges expressed as 16-bit unsigned integers encoding latency values with a resolution of 0.01 ms. This can cover a range of latency values from 0 ms to 655 ms. A properly formatted array specifies bin edges in monotonically increasing order, and operation is undefined otherwise.

D.3.2.2.5 NumSnapshots - 2 bytes

The number of Latency Snapshot Entries for this Service Flow.

D.3.2.2.6 First Snapshot Start Timestamp – 4 bytes

The start time for the first Snapshot Entry for this instance.

D.3.2.2.7 Latency Snapshot Entry Format

Each Latency Snapshot Entry consists of a value for the Snapshot End TimeStamp, an array of Latency Bin Counts (NumPktsBin1, NumPktsBin2, … NumPktsBinN,), a Max Latency value, the number of SanctionedPackets, and the number of histogram updates for a snapshot interval.

- Snapshot End TimeStamp - 4 bytes

The time when this Latency Snapshot ended. The value is expressed in epoch time (also known as 'unix time') and is defined as the number of seconds that have elapsed since midnight Coordinated Universal Time (UTC), Thursday, 1 January 1970.

- Array of Bin counts - (8 bytes * (NumberOfBinEdges + 1))

  An array representing the histogram bin counts during the Snapshot interval for the service flow. Each bin is an 8 Byte value. The length of the array is NumberOfBinEdges+1.

- Maximum Latency - 4 bytes

  The maximum latency (in microseconds) measured for this service flow during the Snapshot interval.

- Number of Histogram Updates - 4 bytes

  The number of histogram updates on this service flow during the snapshot interval.

- Number of SanctionedPackets - 4 bytes

  For an Upstream Low Latency Service Flow, this attribute counts the number of packets redirected from the Low Latency Service Flow to the Classic Service Flow recorded on this Service Flow during the snapshot interval. For other Service Flow types, this counter reports 0.

- Number of Dropped Packets - 4 bytes

  The total number of dropped packets on this service flow during the snapshot interval.

- Number of TotalEct0Pkts Packets - 4 bytes

  This attribute represents the count of packets that arrived marked as ECT0.

- Number of TotalEct1Pkts Packets - 4 bytes

  This attribute represents the count of packets that arrived marked as ECT1.

- Number of CeMarkedEct1Pkts Packets - 4 bytes

  This attribute represents the count of packets that arrived marked as ECT1 and were marked as Congestion Experienced (CE) by the CM.

D.3.3 Example

This is an example showing the US Latency Data encoding for 2 Service Flows. In this example, there are only 5 Snapshots for each service flow, whereas normally there could be more depending on the setting (e.g., 2000 sets of latency data for say 2 second intervals, and 1000 as the number of snapshot intervals).

The columns of data correspond to the values for the Snapshot End TimeStamp, NumPktsBin1, NumPktsBin2, … NumPktsBinN, Max Latency, Number of histogram updates, SanctionedPackets, respectively.

The values in the example are expressed as decimal integer values, but they are encoded using 32-bit hexadecimal notation in the file. The order of latency snapshot entries in the file is from oldest to newest, meaning that the latency snapshot entry following the Header Elements is the oldest set.

Service Flow Id = 1050

SfLabel = 'LLService1'

Number of Bin Edges = 9

Array of Bin Edge definitions = 1,2,5,10,15,20,50,100,400

Number Snapshots = 5

First Snapshot Start Timestamp = 1456252718

TimeStamp	Bin1	Bin2	Bin3	Bin4	Bin5	Bin6	Bin7	Bin8	Bin9	Bin10	Max Latency	Num Hist Updates	Num Sanctioned Pkts	Num Dropped Pkts	Num ECT(0) Pkts	Num CE Marked ECT(0) Pkts	Num ECT(1) Pkts	Num CE Marked ECT(1) Pkts
1456252719	10	2303	233	33	3	1	1	1	0	0	2587000	2575	9	0	0	0	2078	125
1456252720	12	2510	233	50	3	1	1	1	0	0	2622295	2519	10	0	0	0	2113	108
1456252721	12	2320	233	30	3	1	1	1	0	0	2128732	2501	8	0	0	0	2089	95
1456252722	10	2430	233	40	3	1	1	0	0	0	1734235	2708	4	0	0	0	2252	134
1456252723	15	2510	233	50	2	1	1	0	0	0	1582395	2797	2	0	0	0	2375	129

Service Flow Id = 1051

SfLabel = 'CService1'

Number of Bin Edges = 9

Array of Bin Edge definitions = 1,2,5,10,15,20,50,500,650

Number Snapshots = 5

First Snapshot Start Timestamp = 1456252718

TimeStamp	Bin1	Bin2	Bin3	Bin4	Bin5	Bin6	Bin7	Bin8	Bin9	Bin10	Max Latency	Num Hist Updates	Num Sanctioned Pkts	Num Dropped Pkts	Num ECT(0) Pkts	Num CE Marked ECT(0) Pkts	Num ECT(1) Pkts	Num CE Marked ECT(1) Pkts
1456252719	11	11	23	233	2303	33	11	3	1	1	52948304	62	0	8	0	0	0	0
1456252720	13	13	25	220	2510	50	13	5	2	2	71933197	63	0	17	0	0	0	0
1456252721	14	14	23	308	2320	30	14	3	2	1	35531472	62	0	15	0	0	0	0
1456252722	12	12	24	207	2430	40	12	3	1	0	27994462	63	0	4	0	0	0	0
1456252723	16	16	25	217	2510	50	16	2	1	0	28156548	62	0	6	0	0	0	0

Hex Notation:

0000041A 0000000000004c4c5365727669636531 09 0064 00c8 01F4 03E8 05DC 07D0 1388 2710 9c40 0005 56CCA72E

56CCA72F 0000000000000000A 00000000000008FF … 00000039 00000A0F 00000009 00000000 … 0000081E

56CCA730 0000000000000000C 00000000000009CE … 0000003C 000009D7 0000000A 0000000A…00000841

…

…

…

0000041B 0000000000000043536572766963653 1 09 0064 00c8 01F4 03E8 05DC 07D0 1388 C350 FDE8 0005 56CCA72E

56CCA72F 0000000000000000B 000000000000000B … 000003E9 0000003E 00000000 00000008 00000000

56CCA730 0000000000000000D 000000000000000D … 0000041A 0000003F 00000000 00000011 00000000

…

D.4 Slope and Ripple Algorithms

D.4.1 Overview

This section provides details to specify the summary metrics for downstream channel estimate coefficients and upstream pre-equalizer coefficients. The summary metrics are the slope, mean, RMS ripple, and peak-to-peak ripple, for both the magnitude and group delay of the coefficients. The purpose of these summary metrics is to avoid having to send all 2K/4K/8K coefficients on every query. These definitions are intended to replace the less-specific reference in the OSSI spec to the SCTE Measurement Recommended Practices document [SCTE RP].

To provide a basis for the analysis, downstream channel estimate coefficient data was taken from a DOCSIS 3.1 OFDM downstream receiver, with channel start frequency of 543.6 MHz, subcarrier spacing of 50 kHz, and 1880 subcarriers over a bandwidth of 94 MHz. The channel was nominally AWGN with high SNR, although there is some slope and ripple due to the test setup.

D.4.2 Best-Fit Equations

A best-fit line to a set of real-valued data $\{x_i, y_i\}$ may be computed using the following equations, which appear in section 6.5, Nominal Relative Carrier Power Levels and Carrier Level Variations, and Appendix, of [SCTE RP].

$$m = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{N}(x_i - \bar{x})^2}$$

$$b = \bar{y} - m\bar{x}$$

where $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$ and $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$ are the means of x and y, respectively.

The best-fit line has samples

$$y_{fi} = mx_i + b$$

The residual samples after removing the fit line is

$$y_{ri} = y_i - y_{fi}$$

The RMS and peak-to-peak ripple are computed based on the residual values as follows:

$$R_{rms} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} y_{ri}^2}$$

$$R_{pp} = \max_{1 \le i \le N} y_{ri} - \min_{1 \le i \le N} y_{ri}$$

The above equations are general and apply to both the magnitude and group delay responses. The data points $\{x_i, y_i\}$ need not be equally spaced on the x (frequency) axis; that is, the equations continue to be valid if there are missing values of the channel response due to, for example, excluded subcarriers.

D.4.3 Magnitude Summary Metrics

We are given a set of N_C complex channel estimate coefficients

$$C_i = I_i + jQ_i \quad \text{for } i=1 \text{ to } N_c$$

where each channel estimate coefficient corresponds to a subcarrier frequency. The subcarrier frequencies for which channel estimates are provided need not be contiguous. To compute the magnitude summary metrics, the inputs applied to the best-fit equations are N_C, the number of subcarriers for which channel estimate data is provided; x, the list of subcarrier frequencies in MHz; and y, the list of channel estimate coefficient magnitudes in dB for each subcarrier. The y values are computed using

$$y_i = 10\log_{10}(I_i^2 + Q_i^2)$$

For the example data, the log-magnitude-vs-frequency response of the channel estimate coefficients read out from the receiver is plotted in Figure 11 in blue.

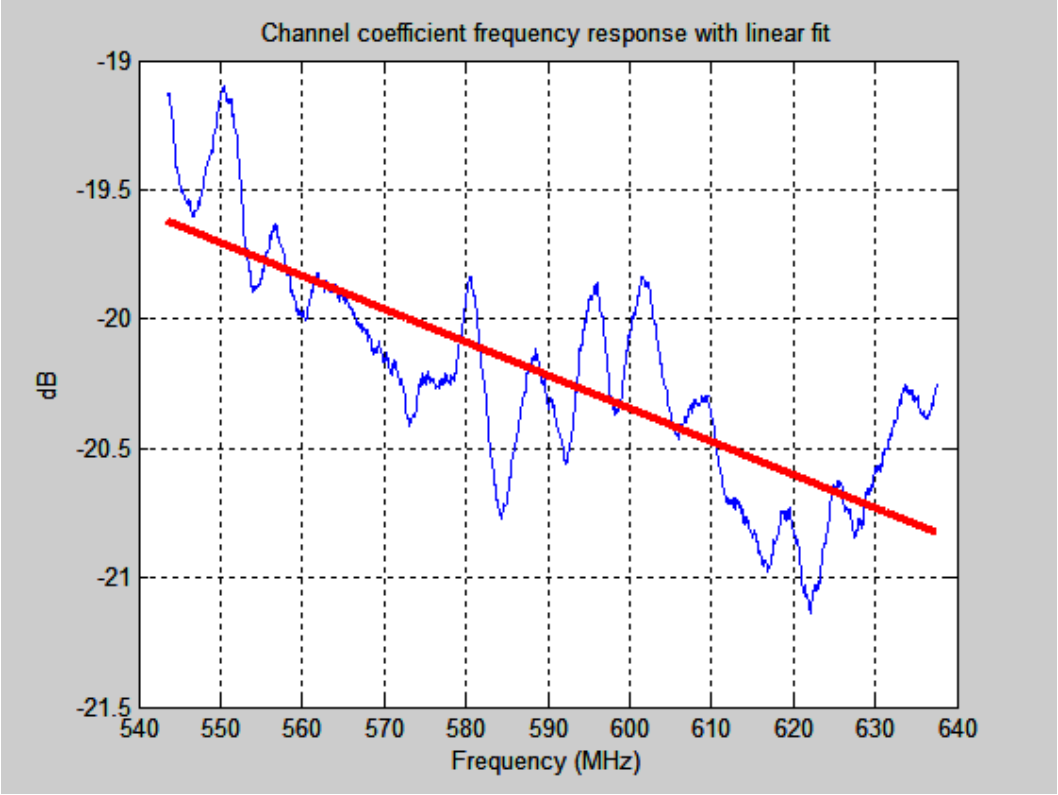


Figure 11 - Frequency Response of Channel Estimate Coefficient, with Best-Fit Line

The calculations for the best-fit line for magnitude are performed using the above best-fit equations directly on the dB values of y. The best-fit line is plotted in Figure 11 in red. The summary metrics for magnitude are the slope m in dB/MHz, mean amplitude $\bar{y} = A_{mean}$ in dB, RMS ripple R_{rms} in dB, and the peak-to-peak ripple R_{pp} in dB.

The metrics for the above example data are

m = -0.0128 dB/MHz

A_{mean} = -20.2224 dB

R_{rms} = 0.2658 dB

R_{pp} = 1.2379 dB

D.4.4 Group Delay Summary Metrics

Group delay is the time delay experienced by a modulated signal, such as a QAM or OFDM/OFDMA downstream or upstream signal, as it passes through a transmission system such as the cable plant. If the group delay is different

for different frequency components across the signal bandwidth, it causes distortion of the signal. Thus, measuring the group delay vs frequency provides important information about plant performance.

Given the N_c complex channel estimate coefficients

$$C_i = I_i + jQ_i \quad \text{for } i=1 \text{ to } N_c$$

the group delay is computed as follows. The angle in radians of each channel coefficient is computed as

$$\phi_i = \arg C_i$$

The general definition for group delay is

$$\tau(\omega) = -\frac{d\phi(\omega)}{d\omega}$$

where τ is the group delay in seconds, ϕ is the unwrapped phase in radians, and ω is the frequency in radians per second. In our case, we are not given a continuous function of radian frequency ω since the channel estimate coefficients are defined at discrete frequencies corresponding to each subcarrier for which channel estimate data is provided. We can approximate the group delay using phase and frequency differences from subcarrier to subcarrier in place of the derivative. This gives the formula for group delay in ns as

$$\tau_i = -\frac{10^9}{2\pi} \cdot \frac{\Delta\phi_i}{\Delta f}$$

where Δf is the subcarrier spacing in Hz, the phase samples are in radians, and the phase difference $\Delta\phi_i$ is defined modulo 2π as

$$\Delta\phi_i = \phi_i - \phi_{i-1} + 2k\pi$$

with the integer k selected such that

$$-\pi \le \Delta\phi_i < \pi$$

This modulo operation on the phase differences effectively unwraps the phase, preventing jumps at 2π interval boundaries that could cause spikes in the group delay.

The frequency value associated with the group delay sample computed on two adjacent subcarriers with respective frequencies f_i and $f_i + \Delta f$ is

$$f_{gi} = f_i + \Delta f / 2$$

that is, halfway between the two adjacent subcarriers.

Since we are using phase differences from subcarrier to subcarrier to compute the group delay, the number of group delay values N_g will be less than the number of channel estimate coefficients N_c. In addition, each isolated missing channel coefficient will cause two missing group delay values. While more complex processing could provide estimates of the group delay at the missing subcarriers, the approach described herein is considered adequate to provide meaningful summary metrics.

Having computed the group delay for each pair of consecutive channel coefficients, we apply the above best-fit equations to compute the summary metrics. The inputs to the best-fit equations for group delay are N_g, the number of group delay values; $x = \{f_{gi}\}$, the list of frequencies in MHz corresponding to the group delay values; and $y = \{\tau_i\}$, the list of group delay values in ns. The outputs are the slope m in ns/MHz, mean group delay $\bar{y} = G_{mean}$ in ns, RMS ripple R_{rms} in ns, and peak-to-peak ripple R_{pp} in ns.

The group-delay-vs-frequency response for the above example channel is plotted in Figure 12 with the best-fit line in red.

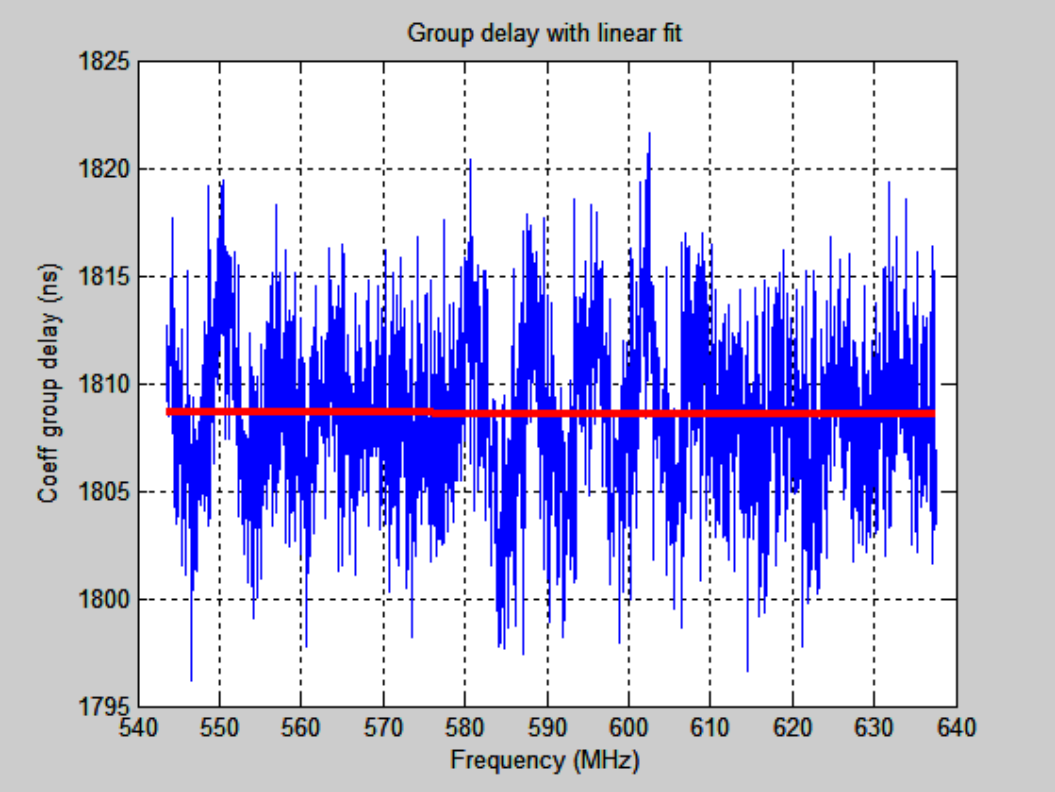


Figure 12 - Group Delay Response Of Channel Estimate Coefficients, With Best-Fit Line

The summary group delay metrics for the example data are

m = -0.0012 ns/MHz

G_{mean} = 1808.6 ns

R_{rms} = 4.0300 ns

R_{pp} = 25.4990 ns

D.4.5 Group Delay Summary Metrics Example

A numerical example of the calculations for the summary metrics is provided in [Slope/Ripple]. The referenced spreadsheet contains an example with channel start frequency of 543.6 MHz, subcarrier spacing of 50 kHz, and 16 channel estimate coefficients corresponding to 16 active subcarriers, plus one excluded subcarrier at 544.05 MHz. The cells in the spreadsheet contain formulas and results for intermediate computations, as well as the final summary metrics (MIB outputs). The second tab in the spreadsheet has plots for amplitude and group delay, including the best-fit line for each.

D.5 Bulk Data Transfer

Proactive Network Maintenance, and potentially other applications, may generate data files that need to be transferred to a server. The Bulk-Data Transfer mechanism defines file storage requirements, destination address and a mechanism to initiate a transfer. The transfer of the bulk data file may be initiated automatically on the file creation or on demand. This section defines the Bulk-Data capability requirements.

D.5.1 CM Bulk Data Transfer Requirements

The CM MUST act as a TFTP client and implement the TFTP protocol over UDP per [RFC 1350] to transfer Bulk-Data files.

The CM MUST initiate the TFTP connection on the standard TFTP-assigned port (69).

The CM MUST use the 'octet' TFTP transfer mode to perform a TFTP 'write' to the specified address.

The CM MUST include the TFTP Blocksize option [RFC 2348] when establishing a TFTP connection.

The CM MUST request a blocksize of 1448 if using TFTP over IPv4. The CM MUST request a blocksize of 1428 if using TFTP over IPv6.

The CM MUST change the value of the UploadStatus attribute in the CmBulkDataFile object to reflect the status of the upload. There are no requirements for the CM to automatically retry the transfer.

D.5.2 CM Data-File and Storage Requirements

The CM MUST be able to store data files that contain at least 64 kilobytes of data. If the CM is commanded to collect data that exceeds its data size capability, the CM MAY stop collecting data and report an error to the application that commanded it to collect data. The application may further report the error if so defined.

The CM MUST be able to store a minimum of four data files.

If the CM is commanded to collect Bulk-Data and the memory allocated for the Bulk-Data file is full, the CM MUST overwrite the oldest data file with the most recently collected data file. The Bulk-Data Transfer mechanism has no notion of file types; hence, it will always replace the oldest file regardless of the type of data contained in the file.

If the oldest data file is currently being uploaded, and this file would need to be deleted to make space for a new data file (i.e., the CM has already stored its maximum limit of files), the CM MUST set the value of the 'PnmCmCtlStatus' object to a value of 'tempReject(4) while the TFTP Upload is in progress. This serves to limit a new test from generating a data file that would overwrite a data file in use.

The CM MUST retain Bulk Data files in the allocated memory unless it is commanded to delete the file, or the file is overwritten with a new file. The CM MAY retain the Bulk Data files across reboot or reset or power cycle.

Figure 13 - Bulk Data Upload Information Model

D.5.3 Bulk Data Objects

This section defines objects that are used to manage the Bulk-Data files that the CM (referred to here as "the device") has been commanded to capture.

D.5.3.1 CmBulkDataControl

This object provides the configuration attributes needed for the device to upload Bulk Data files to a server.

Table 121 - CmBulkDataControl Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
DestIpAddr	InetAddress [RFC 4001]	R/W			"" (empty string)
DestIpAddrType	InetAddressType [RFC 4001]	R/W	unknown(0) ipv4(1) ipv6(2)		unknown
DestPath	AdminString	R/W			"" (empty string)
UploadControl	Enum	R/W	other(1) noAutoupload(2) autoUpload(3)		autoUpload(3)

D.5.3.1.1 DestIpAddr

This attribute represents the IP address of the server to which the bulk data file is to be sent. This attribute is further defined by the DestIpAddrType attribute. The CM MUST NOT allow the value of DestIpAddr to change if the value of PnmCmCtlStatus is any value other than 'ready'.

D.5.3.1.2 DestIpAddrType

This attribute represents the IP address type of the DestIpAddr attribute. This value is of type InetAddressType, which is defined by [RFC 4001].

A successful connection depends on the value of this attribute being set to an IP Family supported by the device. For example, if this value is set to IPv6 and the device is operating in an IPv4-only mode, a successful upload will not be possible. In this case the UploadStatus attribute in the BulkDataFile object would reflect the error. The CM MUST NOT allow the value of DestIpAddrType to change if the value of PnmCmCtlStatus is any value other than 'ready'.

D.5.3.1.3 DestPath

This attribute represents the path, excluding the filename, at the server to which the bulk data file is to be sent. The CM MUST NOT allow the value of DestPath to change if the value of PnmCmCtlStatus is any value other than 'ready'. By default, the value of this object is an empty string. If used, this value includes all expected delimiters. The following examples, excluding the quotes, are valid values:

"/Directory1/directory2/"

"/pnm/"

D.5.3.1.4 UploadControl

This attribute controls the action taken by the device when a new bulk data file is generated. The possible values are defined below.

noAutoUpload(2) - Bulk Data files are not automatically uploaded by the device. All bulk data files are available to be uploaded, on demand, by manipulating the FileControl attribute in the BulkDataFile object for that file's row instance.

autoUpload(3) - When the autoUpload option is selected, the CM MUST automatically upload bulk data files as they become available. A file becomes available when a file-generation application completes the file and creates a row in the BulkDataFileTable. If this value is set, the bulk data file is automatically uploaded to the parameters defined by the DestIpAddr, DestIpAddrType, and DestPath. If the upload fails or additional uploads are desired, the file can be re-uploaded by manipulating the FileControl attribute in the BulkDataFile object for that file's row instance.

D.5.3.2 CmBulkDataFile

This table provides the attributes needed for the device to upload a bulk data file to the Server. This object is a table with a row for each file that is available in the device for upload. The parameters used for the upload are provided under the CmBulkDataControl object.

The CM MUST create a row for each file that is available for upload. The device could have limited resources to save captured data files. Therefore, if the number of files exceeds the minimum supported number of files requirement for the device, newly-created rows can overwrite/replace existing rows as new data files become available. If a bulk data file is no longer available for upload, the CM MUST remove that file's details from the CmBulkDataFile table.

Table 122 - CmBulkDataFile Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
FileIndex	UnsignedByte	Key			
FileName	AdminString	R/O			

Attribute Name	Type	Access	Type Constraints	Units	Default
FileControl	Enum	R/W	other(1) tftpUpload(2) cancelUpload(3) deleteFile (4)		other(1)
FileUploadStatus	Enum	R/O	other(1) availableForUpload(2) uploadInProgress(3) uploadCompleted(4) uploadPending(5) uploadCancelled(6) error(7)		

D.5.3.2.1 FileIndex

This attribute is the key for the table.

D.5.3.2.2 FileName

This attribute contains the name of the bulk data file, stored in the device, that is available to be uploaded to the server. Filenames are defined by the application that creates them.

D.5.3.2.3 FileControl

This attribute controls the action taken by the device regarding the file specified by the FileName attribute. When a value is written to this attribute for a given row instance, the device is required to take that action on the specified bulk data file. The possible actions are:

other(1) - This value is returned when the object is read. This value is not writeable.

tftpUpload(2) -The CM MUST initiate a TFTP-Write to the server with the parameters specified in the 'DestIpAddr', 'DestIpAddrType', and 'DestPath' attributes of the BulkDataControl object. This action will change the value of the UploadStatus attribute to 'uploadInProgress' while the transfer is ongoing. This object can only be set to 'tftpUpload' when the value of the 'UploadStatus' attribute is not set to a value of 'uploadInProgress' for this row OR for any row in the table. This limits the upload process to one upload at a time. This object will return 'inconsistentValue' for this case.

cancelUpload(3) - The CM MUST cancel a pending upload or an upload currently in progress on this bulk data file. The value of the UploadStatus attribute will be changed to 'uploadCancelled'.

deleteFile(4) - The CM MUST delete the file from its memory and from this table. This object cannot be set to deleteFile(4) while an upload is in progress.

D.5.3.2.4 FileUploadStatus

This attribute reflects the status of the bulk data file. The possible values are listed below.

other(1) - Any condition not covered by the other defined values.

availableForUpload(2) - The file is available to be uploaded.

uploadInProgress(3) - The file is currently being uploaded.

uploadCompleted(4) - The file was successfully uploaded.

uploadPending(5) - The file has been selected for upload, but a condition does not allow the upload to take place. The upload will start when the condition blocking uploads has been removed. For example, another upload that is currently in progress could cause this value to be returned.

uploadCancelled(6) - An upload was cancelled before it completed.

error(7) - An error occurred, and the file was not successfully uploaded.

Annex E DOCSIS 4.0 Data Type Definitions (Normative)

E.1 Overview

This specification has requirements for the SNMP protocol for network management functions.

This Annex includes the data type definitions for the information models defined for use in DOCSIS 4.0. The Unified Modeling Language (UML) is used for modeling the management requirements in DOCSIS 4.0. The data types defined in this Annex are mapped for use with SNMP.

Basic UML notation used in this specification and explained in [UML Guidelines].

E.2 Data Type Mapping

XML is becoming the standard for data definition models. With XML data transformations can be done with or without a model (DTD or Schema definition). DTDs and XML schemas provides additional data validation layer to the applications exchanging XML data. There are several models to map formal notation constructs like ASN.1 to XML [ITU-TX.692], UML to XML, or XML by itself can be used for modeling purposes.

Each area of data information interest approaches XML and defines data models and/or data containment structures and data types. Similarly, SNMP took and modified a subset of ASN.1 for defining the Structured Management Information SMIv1 and SMIv2.

Due to the lack of a unified data model and data types for Network Management a neutral model would be appropriated to allow capturing specific requirements and methodologies from existing protocols and allow forward or reverse engineering of those standards like SNMP to the general information model and vice versa.

E.2.1 Data Type Requirements and Classification

The information model has to provide seamless translation for SMIv2 requirements, in particular when creating MIB modules based on the information model, this specification needs to provide full support of [RFC 2578], [RFC 2579] and the clarifications and recommendations of [RFC 4181].

Thus, there are two data type groups defined for modeling purposes and mapping to protocol data notation roundtrip:

1. General Data types
   Required data types to cover all the management syntax and semantic requirement for all OSSI supported data models. In this category are data types defined in SNMP SMIv2 [RFC 2578], [IPDR/XDR], and [IPDR/SSDG].
2. Extended Data types
   Management protocols specialization based on frequent usage or special semantics. Required data types to cover all the syntax requirement for all OSSI supported data models. In this category are SNMP TEXTUAL-CONVENTION clauses [RFC 2579] of mandatory or recommended usage by [RFC 2579] and [RFC 4181] when modeling for SNMP MIB modules.

E.2.2 Data Type Mapping Methodology

The specification "XML Schema Part 2: Data types Second Edition" is based on [ISO11404] which provides a language-independent data types (see XML Schema reference). The mapping proposed below uses a subset of the XML schema data types to cover both SNMP forward and reverse engineering. Any additional protocol being added should be feasible to provide the particular mappings.

SMIv2 has an extensive experience of data types for management purposes, for illustration consider Counter32 and Counter64 SMIv2 types [RFC 2578]. The XML schema data types makes no distinction of derived 'decimal' types and the semantics that are associated to counters, e.g., counters do not necessarily start at 0.

Since the information model needs to cover the mapping of objects to SNMP, the mapping in Section E.2.4 is heavily based on most common SNMP TEXTUAL-CONVENTION descriptors [RFC 2579] and others IETF

commonly used type definitions as well as DOCSIS already defined types in MIB modules required by this specification.

Most of the SNMP information associated to data types are reduced to size and range constraints and specialized enumerations.

E.2.3 General Data Types

Table 123 represents the mapping between the OSSI information model General Types and their equivalent representation for SNMP MIB Modules. The permitted values for the data types are indicated in terms of value ranges and string length when applicable. The IM Data Type column includes the data types to map to SNMP, using the appropriate type in the corresponding protocol if applicable or available. The SNMP Mapping references to SNMP data types are defined in [RFC 2578] or as described below.

Note that SNMP does not provide float, double or long XML-Schema data types. Also, SNMP might map a type to a SNMP subtyped value. For example, UnsignedByte data type maps to Unsigned32 subtyped to the appropriate range indicated by the Permitted Values (0..255 in this case). Other data types are mapped to SNMP TEXTUAL-CONVENTIONS as indicated by the references.

Table 123 - General Data Types

IM Data Type	XML-Schema Data Type	Permitted Values	SNMP Mapping
Boolean	Boolean	true = 1 false = 0	TruthValue [RFC 2579]
Counter32	unsignedInt		Counter32
Counter64	unsignedLong		Counter64
DateTime	dateTime		DateAndTime
Enum	int	-2147483648..2147483647	INTEGER
EnumBits	hexBinary		BITS
HexBinary	hexBinary		OCTET STRING
InetAddress (Deprecated)			InetAddress [RFC 4001]
InetAddressType (Deprecated)			InetAddressType [RFC 4001]
Int	int	-2147483648..2147483647	Integer32
IpAddress		IPv4 Address or IPv6 Address	InetAddress + InetAddressType [RFC 4001]
MacAddress	hexBinary	SIZE (6)	MacAddress
String	string		SnmpAdminString [RFC 3411]
UnsignedByte	unsignedByte	0..255	Unsigned32
UnsignedInt	unsignedInt	0..4294967295	Unsigned32
UnsignedLong	unsignedLong	0..18446744073709551615	CounterBasedGauge64 [RFC 2856]
UnsignedShort	unsignedShort	0..65535	Unsigned32

E.2.4 Extended Data Types

There are two sources of Extended Data Types: Protocol specific data types, and OSSI data types.

SNMP derived types are defined in SNMP MIB Modules. The most important are in [RFC 2579] which is part of SNMP STD 58 and are considered in many aspects part of the SNMP protocol. Other MIB modules TEXTUAL-CONVENTION definitions have been adopted and recommended (e.g., [RFC 4181]) for re-usability and semantics considerations in order to unify management concepts; some relevant RFCs that include common used textual conventions are [RFC 4001], [RFC 2863], [RFC 3411], and [RFC 3419] among others (see [RFC 4181]).

Table 124 includes the most relevant data types taken from SNMP to provide a direct mapping of the OSSI information model to SNMP MIB modules. A few have taken a more general name as they are used across the information models. For example, AdminString comes from [RFC 3411] SnmpAdminString.

In general, when an OSSI information model needs to reference an existing SNMP textual convention for the purpose of round-trip design from UML to SNMP, these textual conventions can be added to this list. Other sources of textual conventions not listed here are from MIB modules specific to DOCSIS either as RFCs or Annex documents in this specification.

OSSI data types are also defined in this specification in the Data Type section of OSSI annexes; for example, Annex A and Annex G.

Table 124 - Extended Data Types

IM Data Type	XML-Schema Data Type	Permitted Values	SNMP Mapping
AdminString	string	SIZE (0..255)	SnmpAdminString
DocsEqualizerData	hexBinary		DocsEqualizerData [RFC 4546]
DocsisUpstreamType	int		DocsisUpstreamType [RFC 4546]
DocsX509ASN1DEREncodedCertificate	hexBinary	SIZE (0..4096)	DocsX509ASN1DEREncodedCertificate [RFC 4131] [RFC 5280]
Duration	unsignedInt	0..2147483647	TimeInterval
InetPortNumber	unsignedInt	0..65535	Unsigned32
PhysAddress	hexBinary		PhysAddress
RowStatus	int		RowStatus
StorageType	int		StorageType
TAddress	hexBinary	SIZE (1..255)	TAddress
TDomain	anyURI		TDomain
TenthdB	int		TenthdB [RFC 4546]
TenthdBmV	int		TenthdBmV [RFC 4546]
TimeStamp	unsignedInt		TimeStamp

E.2.5 Common Terms Shortened

The following table lists common terms which have been shortened to allow shorter SNMP MIB names. These shortened names are desired to be used consistently throughout the information models, SNMP MIBs, and IPDR schemas. However, in some cases it might not be possible to maintain parity with pre-3.0 DOCSIS requirements.

Table 125 - Shortened Common Terms

Original Word	Shortened Word
Address	Addr
Aggregate	Agg
Algorithm	Alg
Application	App
Attribute	Attr
Authorization	Auth
Channel	Ch
Command	Cmd
Config*	Cfg
Control	Ctrl
Default	Def

Original Word	Shortened Word
Destination	Dest
Direction	Dir
Downstream	Ds
Encryption	Encrypt
Equalization	Eq
Frequency	Freq
Group	Grp
Length	Len
Maximum	Max
Minimum	Min
Multicast	Mcast
Provision*	Prov
Receive	Rx
Registration	Reg
Replication	Repl
Request	Req
Resequence	Reseq
Resequencing	Reseq
Response	Rsp
Segment	Sgmt
Sequence	Seq
Service	Svc
ServiceFlow	Sf
Session(s)	Sess
Source	Src
Threshold	Thrshld
Total	Tot
Transmit	Tx
Upstream	Us
* indicates a wildcard	

E.2.5.1 Exceptions

Data types and managed objects do not consistently use the shortened names. Also, the term ServiceFlowId remains unchanged. Service and ServiceFlow are often not shortened to retain backward compatibility with QoS managed objects.

Annex F CM Status Reporting Requirements (Normative)

F.1 Overview

This Annex defines the operational status reporting requirements for a CM.

F.2 CM Operational Status Object Definitions

This section defines the CM configuration and status reporting objects.

F.2.1 Overview

This section defines the configuration and status reporting requirements for the CM. This information is contained in the [MULPIv4.0] and [PHYv4.0] specifications.

F.2.2 Type Definitions

This section defines data types used in this information model.

Table 126 - Data Type Definitions

Data Type Name	Base Type	Permitted Values
CmRegState	Enum	other(1) notReady(2) notSynchronized(3) phySynchronized(4) dsTopologyResolutionInProgress(21) usParametersAcquired(5) rangingInProgress(22) rangingComplete(6) eaeInProgress(14) dhcpv4InProgress(15) dhcpv6InProgress(16) dhcpV4Complete(7) dhcpV6Complete(17) todEstablished(8) securityEstablished(9) configFileDownloadComplete(10) registrationInProgress(18) registrationComplete(11) accessDenied(13) operational(12) bpiInit(19) forwardingDisabled(20) rfMuteAll(23)
DocsisVersion	Enum	other(0) docsis10(1) docsis11(2) docsis20(3) docsis30(4) docsis31(5)
RangingState	Enum	other (1) aborted(2) retriesExceeded(3) success(4) continue(5) timeoutT4(6)
Tlv8	HexBinary	

F.2.2.1 CmRegState

This data type defines the CM connectivity state as reported by the CM.

References: [MULPIv4.0] Cable Modem - CMTS Interaction section.

The enumerated values associated with the CmRegState are:

- other

  'other' indicates any state not described below.

- notReady

  'notReady' indicates that the CM has not started the registration process yet.

- notSynchronized

  'notSynchronized' indicates that the CM has not initiated or completed the synchronization of the downstream physical layer.

- phySynchronized

  'phySynchronized' indicates that the CM has completed the synchronization of the downstream physical layer.

- dsTopologyResolutionInProgress

  'dsTopologyResolutionInProgress' indicates that the CM is attempting to determine its MD-DS-SG.

- usParametersAcquired

  'usParametersAcquired' indicates that the CM has completed the upstream parameters acquisition or have completed the downstream and upstream service groups resolution, whether the CM is registering in a pre-3.0 or a 3.0 CMTS.

- rangingInProgress

  'rangingInProgress' indicates that the CM has initiated the initial ranging process.

- rangingComplete

  'rangingComplete' indicates that the CM has completed initial ranging and received a Ranging Status of success from the CMTS in the RNG-RSP message.

- eaeInProgress

  'eaeInProgress' indicates that the CM has sent an Auth Info message for EAE.

- dhcpv4InProgress

  'dhcpv4InProgress' indicates that the CM has sent a DHCPv4 DISCOVER to gain IP connectivity.

- dhcpv6InProgress

  'dhcpv6InProgress' indicates that the CM has sent an DHCPv6 Solicit message.

- dhcpv4Complete

  'dhcpv4Complete' indicates that the CM has received a DHCPv4 ACK message from the CMTS.

- dhcpv6Complete

  'dhcpv6Complete' indicates that the CM has received a DHCPv6 Reply message from the CMTS.

- todEstablished

  'todEstablished' indicates that the CM has successfully acquired time of day; if the ToD is acquired after the CM is operational, this value SHOULD NOT be reported.

- securityEstablished

  'securityEstablished' indicates that the CM has successfully completed the BPI initialization process.

- configFileDownloadComplete

  'configFileDownloadComplete' indicates that the CM has completed the config file download process.

- registrationInProgress

  'registrationInProgress' indicates that the CM has sent a Registration Request (REG-REQ or REG-REQ-MP)

- registrationComplete

  'registrationComplete' indicates that the CM has successfully completed the Registration process with the CMTS.

- accessDenied

  'accessDenied' indicates that the CM has received a registration aborted notification from the CMTS.

- operational

  'operational' indicates that the CM has completed all necessary initialization steps and is operational.

- bpiInit

  'bpiInit' indicates that the CM has started the BPI initialization process as indicated in the CM config file. If the CM already performed EAE, this state is skipped by the CM.

- forwardingDisabled

  'forwardingDisabled' indicates that the registration process was completed, but the network access option in the received configuration file prohibits forwarding.

- rfMuteAll

  'rfMuteAll' indicates that the CM is instructed to mute all channels in the CM-CTRL-REQ message from CMTS.

The following table provides a mapping of Pre-3.0 DOCSIS and DOCSIS 3.0, 3.1, and 4.0 registration states as reported by CM.

Table 127 - Pre-3.0 DOCSIS and DOCSIS 3.0/3.1/4.0 CM Registration status mapping

CM Pre-3.0 DOCSIS (from docsIfCmStatusValue)	CM DOCSIS 3.0, 3.1 and 4.0
other(1)	other(1)
notReady(2)	notReady(2)
notSynchronized(3)	notSynchronized(3)
phySynchronized(4)	phySynchronized(4)
	dsTopologyResolutionInProgress(21)
usParametersAcquired(5)	usParametersAcquired(5)
	rangingInProgress(22)
rangingComplete(6)	rangingComplete(6)
	eaeInProgress(14)
	dhcpv4InProgress(15)
	dhcpv6InProgress(16)
ipComplete(7)	dhcpv4Complete(7)
	dhcpv6Complete(17)
todEstablished(8)	todEstablished(8)

CM Pre-3.0 DOCSIS (from docsIfCmStatusValue)	CM DOCSIS 3.0, 3.1 and 4.0
securityEstablished(9)	securityEstablished(9)
paramTransferComplete(10)	configFileDownloadComplete(10)
	registrationInProgress(18)
registrationComplete(11)	registrationComplete(11)
accessDenied(13)	accessDenied(13)
operational(12)	operational(12)
	bpiInit (19)
	forwardingDisabled(20)
	rfMuteAll(23)
Note: DOCSIS 3.0 introduced new CM registration states which are given higher enumeration values even though they are intermediate CM registration states.	

F.2.2.2 DocsisVersion

This data type defines the DOCSIS capability of a device.

The enumerated values associated with the DocsisVersion are:

- other

  'other' indicates any state not described below.

- docsis10

  'docsis10' indicates DOCSIS 1.0.

- docsis11

  'docsis11' indicates DOCSIS 1.1.

- docsis20

  'docsis20' indicates DOCSIS 2.0.

- docsis30

  'docsis30' indicates DOCSIS 3.0.

- docsis31

  'docsis31' indicates DOCSIS 3.1.

- docsis40

  'docsis40' indicates DOCSIS 4.0.

F.2.2.3 RangingState

This data type defines the ranging status of the Upstream Channel.

References: [MULPIv4.0] Cable Modem - CMTS Interaction section

The enumerated values associated with the RangingState are:

- Other

  'other' indicates any state not described below.

- Aborted

  'aborted' indicates that the CMTS has sent a ranging abort.

- retriesExceeded

  'retriesExceeded' indicates CM ranging retry limit has been exceeded.

- Success

  'success' indicates that the CMTS has sent a ranging success in the ranging response.

- Continue

  'continue' indicates that the CMTS has sent a ranging continue in the ranging response.

- timeoutT4

  'timeoutT4' indicates that the T4 timer expired on the CM.

F.2.2.4 Tlv8

This data type represents a single TLV encoding. This first octet represents the Type of the TLV. The second octet represents an unsigned 8-bit Length of the subsequent Value part of the TLV. The remaining octets represent the value. The Value could be an atomic value or a sequence of one or more sub-TLVs.

References: [MULPIv4.0] Common Radio Frequency Interface Encodings Annex.

F.2.3 CM Operational Status Objects

These objects report the CM configuration and operational status.

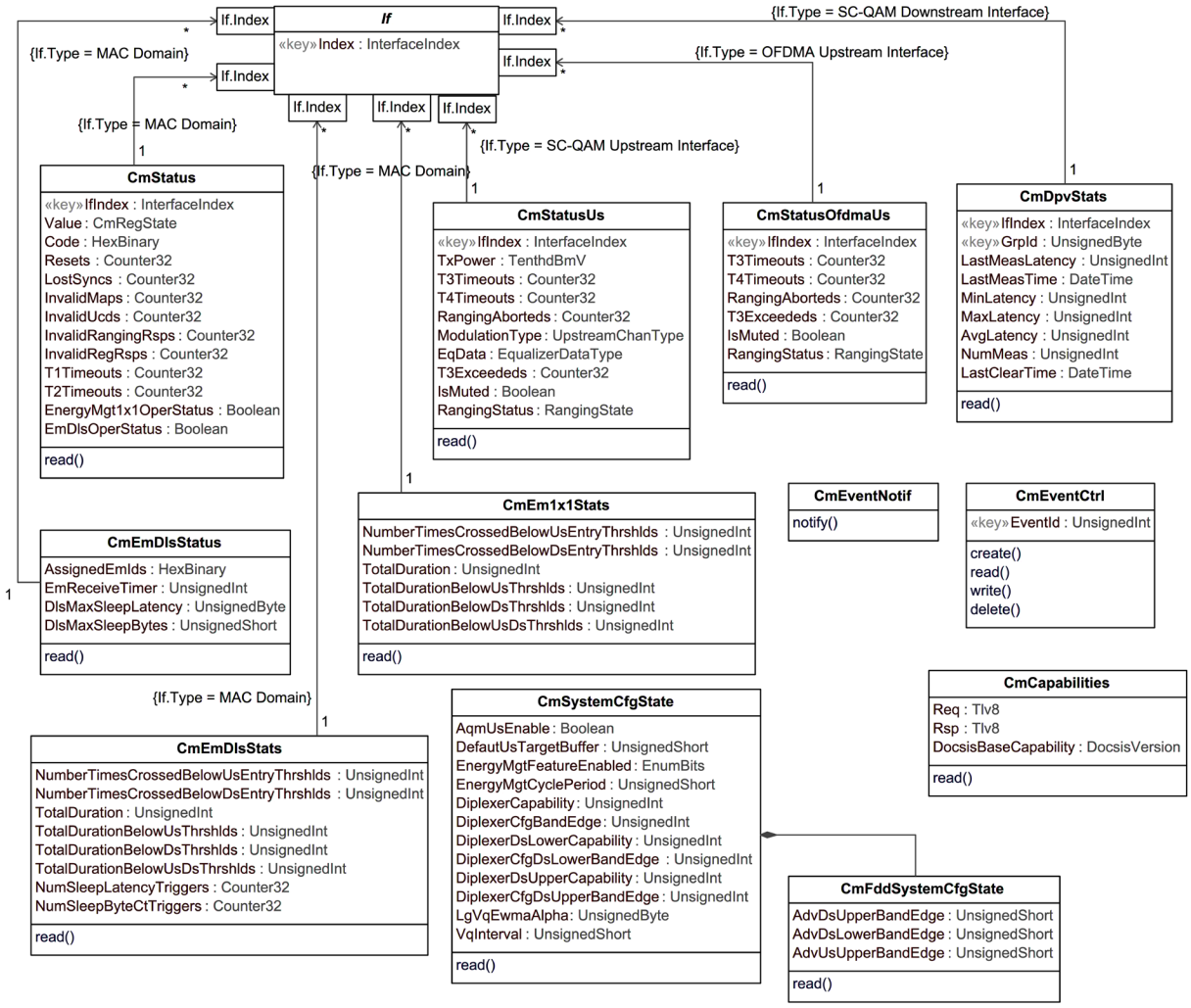


Figure 14 - CM Operational Status Information Model

F.2.3.1 CmStatus

This object provides CM connectivity status information of the CM previously available in the SNMP table docsIfCmStatusTable.

References: [RFC 4546]

Table 128 - CmStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of the MAC interface		
Value	CmRegState	R/O			
Code	HexBinary	R/O	SIZE(0 | 5 | 6)		
Resets	Counter32	R/O		resets	
LostSyncs	Counter32	R/O		messages	
InvalidMaps	Counter32	R/O		maps	
InvalidUcds	Counter32	R/O		messages	
InvalidRangingRsps	Counter32	R/O		messages	

Attribute Name	Type	Access	Type Constraints	Units	Default
InvalidRegRsps	Counter32	R/O		messages	
T1Timeouts	Counter32	R/O		timeouts	
T2Timeouts	Counter32	R/O		timeouts	
EnergyMgt1x1OperStatus	Boolean	R/O			
EmDlsOperStatus	Boolean	R/O			

F.2.3.1.1 IfIndex

This attribute denotes the MAC Domain interface index of the CM.

F.2.3.1.2 Value

This attribute denotes the current CM connectivity state. For the case of IP acquisition related states, this attribute reflects states for the current CM provisioning mode, not the other DHCP process associated with dual stack operation.

References: [MULPIv4.0] Establishing IP Connectivity section

F.2.3.1.3 Code

This attribute denotes the status code for CM as defined in the OSSI Specification. The status code consists of a single character indicating error groups, followed by a two- or three-digit number indicating the status condition, followed by a decimal. An example of a returned value could be 'T101.0'. The zero-length hex string indicates no status code yet registered.

References: Annex C

F.2.3.1.4 Resets

This attribute denotes the number of times the CM reset or initialized this interface. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.5 LostSyncs

This attribute denotes the number of times the CM lost synchronization with the downstream channel. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.6 InvalidMaps

This attribute denotes the number of times the CM received invalid MAP messages. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.7 InvalidUcds

This attribute denotes the number of times the CM received invalid UCD messages. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.8 InvalidRangingRsps

This attribute denotes the number of times the CM received invalid ranging response messages. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.9 InvalidRegRsps

This attribute denotes the number of times the CM received invalid registration response messages. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

F.2.3.1.10 T1Timeouts

This attribute denotes the number of times counter T1 expired in the CM. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.11 T2Timeouts

This attribute denotes the number of times counter T2 expired in the CM. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the CM MAC Domain interface.

References: [RFC 2863]

F.2.3.1.12 EnergyMgt1x1OperStatus

This attribute indicates whether the CM is currently operating in Energy Management 1x1 Mode. If this attribute returns true, the CM is operating in Energy Management 1x1 Mode.

References: [MULPIv4.0] Energy Management Mode Indicator section.

F.2.3.1.13 EmDlsOperStatus

This attribute indicates whether the CM is currently operating in Energy Management DLS Mode. If this attribute returns true, the CM is operating in Energy Management DLS Mode.

References: [MULPIv4.0] Energy Management Mode Indicator section.

F.2.3.2 CmStatusUs

This object defines PHY and MAC information about the CM's SC-QAM upstream channels. This object provides per-CM Upstream channel information previously available in the SNMP table docsIfCmStatusTable.

Table 129 - CmStatusUs Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of upstream interface		
TxPower	TenthdBmV	R/O		dBmV	
T3Timeouts	Counter32	R/O		timeouts	
T4Timeouts	Counter32	R/O		timeouts	
RangingAborteds	Counter32	R/O		attempts	
ModulationType	DocsisUpstreamType	R/O			
EqData	DocsEqualizerData	R/O			
T3Exceededs	Counter32	R/O		timeouts	

Attribute Name	Type	Access	Type Constraints	Units	Default
IsMuted	Boolean	R/O			
RangingStatus	RangingState	R/O			

F.2.3.2.1 IfIndex

This attribute denotes the interface index of the upstream interface to which this instance applies.

F.2.3.2.2 TxPower

This attribute denotes the operational CM transmit power for this SC-QAM upstream channel. In order for this attribute to provide consistent information under all circumstances, a 4.0 CM will report the average total power for the SC-QAM channel the same as was done for DOCSIS 3.0, regardless of whether it is operating with a 4.0, 3.1, or a 3.0 CMTS. The value that is reported was referred to as Pr in [PHYv3.0].

F.2.3.2.3 T3Timeouts

This attribute denotes the number of times counter T3 expired in the CM for this upstream channel. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.2.4 T4Timeouts

This attribute denotes the number of times counter T4 expired in the CM for this upstream channel. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.2.5 RangingAborteds

This attribute denotes the number of times the ranging process was aborted by the CMTS. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.2.6 ModulationType

This attribute denotes the modulation type status currently used by the CM for this upstream channel. Since this object specifically identifies PHY Layer mode, the shared upstream channel type 'tdmaAndAtdma' is not permitted.

References: [RFC 2863]

F.2.3.2.7 EqData

This attribute denotes the pre-equalization data for the specified upstream channel on this CM after convolution with data indicated in the RNG-RSP. This data is valid when docsIfUpChannelPreEqEnable is set to 'true'.

References: [RFC 4546]

F.2.3.2.8 T3Exceededs

This attribute denotes the number of times for excessive T3 timeouts. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.2.9 IsMuted

This attribute denotes whether the upstream channel is muted.

References: [MULPIv4.0] Media Access Control Specification section

F.2.3.2.10 RangingStatus

This attribute denotes ranging status of this upstream channel.

References: [MULPIv4.0] Media Access Control Specification section

F.2.3.3 CmStatusOfdmaUs

This object defines PHY and MAC information about the CM's OFDMA upstream channels. This object provides per-CM Upstream channel information previously available in the SNMP table docsIfCmStatusTable.

Table 130 - CmStatusOfdmaUs Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of upstream interface		
T3Timeouts	Counter32	R/O		timeouts	
T4Timeouts	Counter32	R/O		timeouts	
RangingAborteds	Counter32	R/O		attempts	
T3Exceededs	Counter32	R/O		timeouts	
IsMuted	Boolean	R/O			
RangingStatus	RangingState	R/O			

F.2.3.3.1 IfIndex

This attribute denotes the interface index of the upstream interface to which this instance applies.

F.2.3.3.2 T3Timeouts

This attribute denotes the number of times counter T3 expired in the CM for this upstream channel. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.3.3 T4Timeouts

This attribute denotes the number of times counter T4 expired in the CM for this upstream channel. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.3.4 RangingAborteds

This attribute denotes the number of times the ranging process was aborted by the CMTS. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.3.5 T3Exceededs

This attribute denotes the number of times for excessive T3 timeouts. Discontinuities in the value of this counter can occur at re-initialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated upstream channel.

References: [RFC 2863]

F.2.3.3.6 IsMuted

This attribute denotes whether the upstream channel is muted.

References: [MULPIv4.0] Media Access Control Specification section

F.2.3.3.7 RangingStatus

This attribute denotes ranging status of this upstream channel.

References: [MULPIv4.0] Media Access Control Specification section

F.2.3.4 CmCapabilities

This object defines attributes of the CM capabilities.

Table 131 - CmCapabilities Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Req	Tlv8	R/O			
Rsp	Tlv8	R/O			
DocsisBaseCapability	DocsisVersion	R/O			

F.2.3.4.1 Req

This attribute contains the TLV encoding for TLV-5 sent in a REG-REQ. The first byte of this encoding is expected to be '05'H.

References: [MULPIv4.0] Modem Capabilities Encoding section in the Common Radio Frequency Interface Encodings Annex

F.2.3.4.2 Rsp

This attribute contains the TLV encoding for TLV-5 (see the Modem Capabilities Encoding section in Common Radio Frequency Interface Encodings Annex of) received in a REG-RSP. The first byte of this encoding is expected to be '05'H.

References: [MULPIv4.0] Modem Capabilities Encoding section in the Common Radio Frequency Interface Encodings Annex

F.2.3.4.3 DocsisBaseCapability

This attribute indicates the DOCSIS capability of the CM. CMs report their supported DOCSIS version. This attribute replaces docsIfDocsisBaseCapability defined in [RFC 4546].

F.2.3.5 CmDpvStats

This object represents the DOCSIS Path Verify Statistics collected in the cable modem device. The CMTS controls the logging of DPV statistics in the cable modem. Therefore, the context and nature of the measurements are governed by the CMTS and not self-descriptive when read from the CM.

Table 132 - CmDpvStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface index of downstream interface		
GrpId	UnsignedByte	Key	1..2		
LastMeasLatency	UnsignedInt	R/O		nanoseconds	
LastMeasTime	DateTime	R/O			
MinLatency	UnsignedInt	R/O		nanoseconds	

Attribute Name	Type	Access	Type Constraints	Units	Default
MaxLatency	UnsignedInt	R/O		nanoseconds	
AvgLatency	UnsignedInt	R/O		nanoseconds	
NumMeas	UnsignedInt	R/O		nanoseconds	
LastClearTime	DateTime	R/O			

F.2.3.5.1 ifIndex

This key represents the interface Index of the Downstream Interface where the measurements are taken.

F.2.3.5.2 GrpId

This key represents the DPV Group ID. The CM reports two instance of DPV statistics per downstream normally referred as Statistical Group 1 and Statistical Group 2.

F.2.3.5.3 LastMeasLatency

This attribute represents the last latency measurement for this statistical group.

F.2.3.5.4 LastMeasTime

This attribute represents the last measurement time of the last latency measurement for this statistical group. This attribute reports the epoch time value when no measurements are being reported or after the statistics were cleared.

F.2.3.5.5 MinLatency

This attribute represents the minimum latency measurement for this statistical group since the last time statistics were cleared.

F.2.3.5.6 MaxLatency

This attribute represents the maximum latency measurement for this statistical group since the last time statistics were cleared.

F.2.3.5.7 AvgLatency

This attribute represents the average latency measurement for this statistical group since the last time statistics were cleared. The averaging mechanism is controlled by the CMTS.

References: [MULPIv4.0] DPV Math section

F.2.3.5.8 NumMeas

This attribute represents the number of latency measurements made for this statistical group since the last time statistics were cleared.

F.2.3.5.9 LastClearTime

This attribute represents the last time statistics were cleared for this statistical group, otherwise this attribute reports the epoch time value.

F.2.3.6 CmEventCtrl

This object represents the control mechanism to enable the dispatching of events based on the event Id. The following rules define the event control behavior:

- The CmEventCtrl object has no instances or contains an instance with Event ID 0.
  All events matching the Local Log settings of docsDevEvReporting are sent to local log ONLY.

- Additionally, if The CmEventCtrl object contains configured instances with non-zero Event IDs. Events matching the Event Ids configured in the object are sent according to the settings of the docsDevEvReporting object; i.e., Traps, Syslog, etc.

The CM MUST NOT persist instances of CmEventCtrl across reinitializations.

Table 133 - CmEventCtrl Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
EventId	UnsignedInt	Key			

F.2.3.6.1 EventId

This key represents the Event ID of the event being enabled for delivery to a dispatch mechanism (e.g., syslog).

References: Annex C

F.2.3.7 CmEventNotif

This object represents the abstract definition of an event object for the CM. The realization of the event object depends of the management protocol that carries the event. For example, the object event realization as a SNMP notification is defined as the docsIf3CmEventNotif defined in [DOCS-IF3-MIB].

F.2.3.8 CmEm1x1Stats

This object defines Energy Management 1x1 mode statistics on the CM to provide insight into configuration of appropriate EM 1x1 Mode Activity Detection thresholds and/or to get feedback on how/if the current thresholds are working well or are causing user experience issues. These statistics are only applicable/valid when the Energy Management 1x1 mode is enabled in the CM.

Table 134 - CmEm1x1Stats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
NumberTimesCrossedBelowUsEntryThrshlds	UnsignedInt	R/O			
NumberTimesCrossedBelowDsEntryThrshlds	UnsignedInt	R/O			
TotalDuration	UnsignedInt	R/O		seconds	
TotalDurationBelowUsThrshlds	UnsignedInt	R/O		seconds	
TotalDurationBelowDsThrshlds	UnsignedInt	R/O		seconds	
TotalDurationBelowUsDsThrshlds	UnsignedInt	R/O		seconds	

F.2.3.8.1 NumberTimesCrossedBelowUsEntryThrshlds

This attribute indicates the number of times since registration the CM crossed below the upstream entry bitrate threshold for a number of consecutive seconds equal to or exceeding the upstream entry time threshold.

F.2.3.8.2 NumberTimesCrossedBelowDsEntryThrshlds

This attribute indicates the number of times since registration the CM crossed below the downstream entry bitrate threshold for a number of consecutive seconds equal to or exceeding the downstream entry time threshold.

F.2.3.8.3 TotalDuration

This attribute indicates the total time duration, in seconds since registration, the CM has been in Energy Management 1x1 mode, as controlled by the DBC-REQ Energy Management 1x1 Mode Indicator TLV. This attribute differs from TotalDurationBelowUsDsThrshlds because it is dependent on effects of the Energy Management Cycle Period, and processing of EM-REQ/EM-RSP messages and DBC messages that specifically indicate entry into or exit from Energy Management 1x1 mode.

F.2.3.8.4 TotalDurationBelowUsThrshlds

This attribute indicates the total time duration, in seconds since registration, the CM satisfied upstream conditions for entry into or remaining in Energy Management 1x1 mode.

F.2.3.8.5 TotalDurationBelowDsThrshlds

This attribute indicates the total time duration, in seconds since registration, the CM satisfied downstream conditions for entry into or remaining in Energy Management 1x1 mode.

F.2.3.8.6 TotalDurationBelowUsDsThrshlds

This attribute indicates the total time duration, in seconds since registration, the CM, with respect to both upstream and downstream entry and exit thresholds, satisfied conditions for entry into and remaining in Energy Management 1x1 mode. This attribute differs from TotalDuration because it is not dependent on effects of the Energy Management Cycle Period or processing of EM-REQ/EM-RSP messages and DBC messages that specifically indicate entry into or exit from Energy Management 1x1 mode.

F.2.3.9 CmEmDlsStats

This object defines the DLS Energy Management mode statistics on the CM to provide insight into configuration of appropriate DLS EM Mode Activity Detection thresholds and/or to get feedback on how/if the current thresholds are working well or are causing user experience issues. These statistics are only applicable/valid when the DLS Energy Management mode is enabled in the CM.

Table 135 - CmEmDlsStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
NumberTimesCrossedBelowUsEntryThrshlds	UnsignedInt	R/O			
NumberTimesCrossedBelowDsEntryThrshlds	UnsignedInt	R/O			
TotalDuration	UnsignedInt	R/O		seconds	
TotalDurationBelowUsThrshlds	UnsignedInt	R/O		seconds	
TotalDurationBelowDsThrshlds	UnsignedInt	R/O		seconds	
TotalDurationBelowUsDsThrshlds	UnsignedInt	R/O		seconds	
NumSleepLatencyTriggers	Counter32	R/O			
NumSleepByteCtTriggers	Counter32	R/O			

F.2.3.9.1 NumberTimesCrossedBelowUsEntryThrshlds

This attribute indicates the number of times since registration the CM crossed below the upstream entry bitrate threshold for a number of consecutive seconds equal to or exceeding the upstream entry time threshold.

F.2.3.9.2 NumberTimesCrossedBelowDsEntryThrshlds

This attribute indicates the number of times since registration the CM crossed below the downstream entry bitrate threshold for a number of consecutive seconds equal to or exceeding the downstream entry time threshold.

F.2.3.9.3 TotalDuration

This attribute indicates the total time duration, in seconds since registration, the CM has been in DLS Energy Management mode, as controlled by the DBC-REQ Energy Management Mode Indicator TLV. This attribute differs from TotalDurationBelowUsDsThrshlds because it is dependent on effects of the Energy Management Cycle Period, and processing of EM-REQ/EM-RSP messages and DBC messages that specifically indicate entry into or exit from DLS Energy Management mode.

F.2.3.9.4 TotalDurationBelowUsThrshlds

This attribute indicates the total time duration, in seconds since registration, the CM satisfied upstream conditions for entry into or remaining in DLS Energy Management mode.

F.2.3.9.5 TotalDurationBelowDsThrshlds

This attribute indicates the total time duration, in seconds since registration, the CM satisfied downstream conditions for entry into or remaining in DLS Energy Management mode.

F.2.3.9.6 TotalDurationBelowUsDsThrshlds

This attribute indicates the total time duration, in seconds since registration, the CM, with respect to both upstream and downstream entry and exit thresholds, satisfied conditions for entry into and remaining in DLS Energy Management mode. This attribute differs from TotalDuration because it is not dependent on effects of the Energy Management Cycle Period or processing of EM-REQ/EM-RSP messages and DBC messages that specifically indicate entry into or exit from DLS Energy Management mode.

F.2.3.9.7 NumSleepLatencyTriggers

This attribute indicates the number of times since registration the CM transitioned to the DLS wake state due to the DLS Maximum Sleep Latency being exceeded.

F.2.3.9.8 NumSleepByteCtTriggers

This attribute indicates the number of times since registration the CM transitioned to the DLS wake state due to the DLS Maximum Byte Count being exceeded.

F.2.3.10 CmEmDlsStatus

This object defines the DLS Energy Management mode status on the CM to provide insight into the current configuration of DLS Mode. This information is only applicable/valid when the DLS Energy Management mode is enabled on the CM.

Table 136 - CmEmDlsStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
AssignedEmIds	HexBinary	R/O	SIZE (2 | 4 | 6)		
DlsReceiveTimer	PlcFrameInterval(1 byte)	R/O	0..2	PlcFrame	0
DlsMaxSleepLatency	UnsignedByte	R/O	1..255	msec	100
DlsMaxSleepBytes	UnsignedShort	R/O	1..65535	bytes	1Kbytes

F.2.3.10.1 AssignedEmIds

This attribute reports the set of CMTS-assigned EM-IDs for this cable modem. This attribute is encoded as an array 16-bit binary values with up to 3 elements. The broadcast EM-ID is not included in the list. This is generally displayed as a comma-delimited list of EM-IDS such as: DF13,ABAB,0002.

References: [MULPIv4.0] DOCSIS Light Sleep Feature section.

F.2.3.10.2 DlsReceiveTimer

This attribute specifies how long the CM is required to continue listening on the downstream for traffic, after reception of the EMM with Sleep Time with a non-zero value. The CMTS communicates the EM Receive Timer to the CM during registration or in DBC message.

F.2.3.10.3 DlsMaxSleepLatency

This attribute specifies the amount of time the CM would allow an upstream channel to queue the packets without transitioning to DLS wake state.

F.2.3.10.4 DlsMaxSleepBytes

This attribute specifies the maximum number of bytes a CM would allow an upstream service flow to enqueue without transitioning to DLS wake state.

F.2.3.11 CmSystemCfgState

The CmSystemCfgState object defines configuration state at the global or system wide level for the CM.

Table 137 - CmSystemCfgState Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
AqmUsEnable	Boolean	R/O			
DefaultUsTargetBuffer	UnsignedShort	R/O		msec	
EnergyMgtFeatureEnabled	EnumBits	R/O	em1x1Feature(0) dls(1)		
EnergyMgtCyclePeriod	UnsignedShort	R/O		seconds	900
DiplexerCapability	UnsignedInt	R/O			
DiplexerCfgBandEdge	UnsignedInt	R/O			
DiplexerDsLowerCapability	UnsignedInt	R/O			
DiplexerCfgDsLowerBandEdge	UnsignedInt	R/O			
DiplexerDsUpperCapability	UnsignedInt	R/O			
DiplexerCfgDsUpperBandEdge	UnsignedInt	R/O			
LgVqEwmaAlpha	UnsignedByte	R/W R/O			7
VqInterval	UnsignedShort	R/W R/O	300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000	microseconds	500

Table 138 - CmSystemCfgState Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
CmFddSystemCfgState	Composition	1	1	

F.2.3.11.1 AqmUsEnable

If this attribute is set to 'false', the CM disables Active Queue Management (AQM) on all upstream service flows.

Reference: [DOCS-QOS3-MIB] docsQosCmSystemCfgStateAqmUsEnable

F.2.3.11.2 DefaultUsTargetBuffer

This attribute specifies the default upstream service flow target buffer size, in milliseconds, when not specified otherwise in service flow TLV encodings.

References: [MULPIv4.0] Default Upstream Target Buffer Configuration Annex; [DOCS-QOS3-MIB] docsQosCmSystemCfgStateDefaultUsTargetBuffer

F.2.3.11.3 EnergyMgtFeatureEnabled

This attribute indicates which energy savings features have been enabled in the Cable Modem. The CM enables use of Energy Management Features only if both the Energy Management Feature Control TLV and Energy Management Modem Capability Response from the CMTS indicate that the feature is enabled. If bit 0 is set, the Energy Management 1x1 Mode feature is enabled. If bit 1 is set, the DOCSIS Light Sleep Mode feature is enabled.

References: [MULPIv4.0] Energy Management Feature Control section; [DOCS-IF3-MIB] docsIf3CmEnergyMgtCfgFeatureEnabled

F.2.3.11.4 EnergyMgtCyclePeriod

This attribute specifies a minimum time period (in seconds) that needs to elapse between EM-REQ transactions in certain situations:

- This attribute sets the minimum cycle time that a CM will use for sending requests to enter an Energy Management Mode. The CM will not request to enter an Energy Management Mode while this amount of time has yet to elapse since the last time the CM requested an Energy Management Mode and received a response indicating (0) OK or (1) Reject Temporary (with no Hold-off Timer value provided).
- In the case that the CM fails to receive an EM-RSP message after the maximum number of retries, this attribute sets the minimum amount of time to elapse before the CM can attempt another EM-REQ transaction.

References: [MULPIv4.0] Energy Management Cycle Period section; [DOCS-IF3-MIB] docsIf3CmEnergyMgtCfgCyclePeriod

F.2.3.11.5 DiplexerCapability

This attribute specifies the upstream diplexer upper band edge configurations supported by the CM device. This corresponds to the Diplexer Upstream Upper Band Edge modem capability sent by the CM in the Registration Request. The Integer value represents the value sent in the Registration Request as a bit mask.

Example:

CM supports Switchable Upstream Diplexer configurations as follows:

```
Bit #0: Upstream Frequency Range up to 42 MHz

Bit #2: Upstream Frequency Range up to 85 MHz
```

The Integer value set for the DiplexerCapability in this case would equal 5.

References: [MULPIv4.0] Diplexer Upstream Upper Band Edge; [DOCS-IF31-MIB] docsIf31CmSystemCfgStateDiplexerCapability

F.2.3.11.6 DiplexerCfgBandEdge

This attribute specifies the current configured frequency of the upstream upper band edge of the diplexer in the CM device. This value is expressed in MHz and corresponds to one of the Diplexer Upper Band Edge capabilities reported by the CM in its Registration Request.

References: [MULPIv4.0] Diplexer Upstream Upper Band Edge; [DOCS-IF31-MIB] docsIf31CmSystemCfgStateDiplexerCfgBandEdge

F.2.3.11.7 DiplexerDsLowerCapability

This attribute specifies the downstream diplexer lower band edge configurations supported by the CM device. This corresponds to the Diplexer Downstream Lower Band Edge modem capability sent by the CM in the Registration Request. The Integer value represents the value sent in the Registration Request as a bit mask.

Example:

CM supports Switchable Downstream Diplexer Lower Band Edge configurations as follows:

```
Bit #0: Downstream Frequency Range starting from 108 MHz

Bit #1: Downstream Frequency Range starting from 258 MHz
```

The Integer value set for the DiplexerDsLowerCapability in this case would be 3.

References: [MULPIv4.0] Diplexer Downstream Lower Band Edge; [DOCS-IF31-MIB] docsIf31CmSystemCfgStateDiplexerDsLowerCapability

F.2.3.11.8 DiplexerCfgDsLowerBandEdge

This attribute specifies the current configured frequency of the downstream lower band edge of the diplexer in the CM device. This value is expressed in MHz and corresponds to one of the Downstream Lower Band Edge capabilities reported by the CM in its Registration Request.

References: [MULPIv4.0] Downstream Lower Band Edge; [DOCS-IF31-MIB] docsIf31CmSystemCfgStateDiplexerCfgDsLowerBandEdge

F.2.3.11.9 DiplexerDsUpperCapability

This attribute specifies the downstream diplexer upper band edge configurations supported by the CM device. This corresponds to the Diplexer Downstream Upper Band Edge modem capability sent by the CM in the Registration Request. The Integer value represents the value sent in the Registration Request as a bit mask.

Example:

CM supports only a single configuration for the DiplexerDsUpperCapability

```
Bit #0: Downstream Frequency Range up to 1218 MHz
```

The Integer values set for the DiplexerDsUpperCapability in this case would be 1.

References: [MULPIv4.0] Diplexer Downstream Upper Band Edge; [DOCS-IF31-MIB] docsIf31CmSystemCfgStateDiplexerDsUpperCapability

F.2.3.11.10 DiplexerCfgDsUpperBandEdge

This attribute specifies the current configured frequency of the downstream upper band edge of the diplexer in the CM device. This value is expressed in MHz and corresponds to one of the Downstream Upper Band Edge capabilities reported by the CM in its Registration Request.

References: [MULPIv4.0] Downstream Upper Band Edge; [DOCS-IF31-MIB] docsIf31CmSystemCfgStateDiplexerCfgDsUpperBandEdge

F.2.3.11.11 LgVqEwmaAlpha

This attribute represents the log base 2 of the reciprocal of the exponentially weighted moving average weight for the Low Latency Virtual Queue (VQ).

This attribute is configurable only through the CM configuration file via TLV-11. This attribute is read-only during the CM operational state.

References: [MULPIv3.1] LG_VQ_EWMA_ALPHA parameter defined in pseudocode in Annex O.1 AQM Utility Functions

F.2.3.11.12 VqInterval

This attribute represents the interval between alignments of the Virtual Queue (VQ) with the actual queue.

This attribute is configurable only through the CM configuration file via TLV-11. This attribute is read-only during the CM operational state.

References: [MULPIv3.1] VQ_INTERVAL parameter defined in pseudocode in Annex O.1 AQM Utility Functions

F.2.3.12 CmFddSystemCfgState

This object reports the configuration properties of Extended Spectrum attributes for the CM.

Table 139 - CmFddSystemCfgState Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
AdvDsLowerBandEdge	UnsignedShort	R/O		MHz	
AdvDsUpperBandEdge	UnsignedShort	R/O		MHz	

Attribute Name	Type	Access	Type Constraints	Units	Default
AdvUsUpperBandEdge	UnsignedShort	R/O		MHz	

F.2.3.12.1 AdvDsLowerBandEdge

This attribute specifies the starting (lowest) frequency for which the downstream band is currently configured in the cable modem. This corresponds to the Advanced Downstream Lower Band Edge Configuration sent by the CM in the Registration Request as TLV type 5.79. The two-byte unsigned integer represents the frequency in MHz. The value zero indicates the CM is currently not configured with a channel in the extended spectrum.

The downstream lower band edge limit for the plant depends on the mode of operation to which DOCSIS equipment is configured to operate. Refer to [PHYv3.1] *Downstream CM Spectrum* section, for the downstream boundary frequency limits for the plant when equipment is configured to be compliant with the DOCSIS 3.1 and (non-FDX and non-FDD extended spectrum) DOCSIS 4.0 frequency plans. Refer to [PHYv4.0] *Downstream FDX CM Spectrum* section for the downstream boundary frequency limits for the plant when equipment is configured to be compliant with FDX mode. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan for FDD Operation* section for the downstream boundary frequency limits for the plant when the equipment is configured to be compliant with FDD mode.

Reference: [MULPIv4.0] Advanced Downstream Lower Band Edge Configuration, [PHYv4.0] Downstream CM Spectrum

F.2.3.12.2 AdvDsUpperBandEdge

This attribute specifies the ending (highest) frequency for which the downstream band is currently configured in the cable modem. This corresponds to the Advanced Downstream Upper Band Edge Configuration sent by the CM in the Registration Request as TLV type 5.80. The two-byte unsigned integer represents the frequency in MHz. The value zero indicates the CM is currently not configured with a channel in the extended spectrum.

The downstream upper band edge limit for the plant depends on the mode of operation to which DOCSIS equipment is configured to operate. Refer to [PHYv3.1] *Downstream CM Spectrum* section, for the downstream boundary frequency limits for the plant when equipment is configured to be compliant with the DOCSIS 3.1 and (non-FDX and non-FDD extended spectrum) DOCSIS 4.0 frequency plans. Refer to [PHYv4.0] *Downstream FDX CM Spectrum* section for the downstream boundary frequency limits for the plant when equipment is configured to be compliant with FDX mode. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan for FDD Operation* section for the downstream boundary frequency limits for the plant when the equipment is configured to be compliant with FDD mode.

Reference: [MULPIv4.0] Advanced Downstream Upper Band Edge Configuration, [PHYv4.0] Downstream CM Spectrum

F.2.3.12.3 AdvUsUpperBandEdge

This attribute specifies the ending (highest) frequency for which the upstream band is currently configured in the cable modem. This corresponds to the Advanced Diplexer Upstream Upper Band Edge Configuration sent by the CM in the Registration Request as TLV type 5.81. The two-byte unsigned integer represents the frequency in MHz. The value zero indicates the CM is currently not configured with a channel in the extended spectrum.

The upstream upper band edge limit for the plant depends on the mode of operation to which DOCSIS equipment is configured to operate. Refer to [PHYv3.1] *Upstream CM Spectrum* section, for the upstream boundary frequency limits for the plant when equipment is configured to be compliant with the DOCSIS 3.1 and (non-FDX and non-FDD extended spectrum) DOCSIS 4.0 frequency plans. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan* section for the upstream boundary frequency limits for the plant when equipment is configured to be compliant with FDX mode. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan for FDD Operation* section for the upstream boundary frequency limits for the plant when the equipment is configured to be compliant with FDD mode.

Reference: [MULPIv4.0] Advanced Diplexer Upstream Upper Band Edge Configuration, [PHYv4.0] Upstream CM Spectrum

F.2.3.13 CmService

This object is only applicable in the DOCSIS 1.0 CoS mode and is therefore deprecated for DOCSIS 4.0.

F.2.3.14 QosProfile

This object is only applicable in the DOCSIS 1.0 CoS mode and is therefore deprecated for DOCSIS 4.0.

F.2.3.15 (Pre-3.0 DOCSIS) CmStatus

This object has been reconstituted as CmStatus and CmStatusUs in DOCSIS 3.0 and is only applicable in the Pre-3.0 DOCSIS versions and is therefore deprecated.

F.3 CM Downstream and Upstream Interfaces Information Models

F.3.1 DS US Common Data Type Definitions

Table 140 - CM Downstream Parameter Data Types

Data Type Name	Base Type	Permitted Values
SubcarrierSpacingType	UnsignedByte	(25 | 50)

F.3.1.1 SubcarrierSpacingType

This data type defines the subcarrier spacing for the FFT mode in use. For downstream OFDM channels, if the FFT mode is 4K mode, then spacing is 50 kHz; if it is 8K mode, then the spacing is 25 kHz. These values of subcarrier spacing (for downstream OFDM channels) are defined in the Downstream OFDM Parameters table in [PHYv4.0]. For upstream OFDMA channels, if the FFT mode is 2K mode, then the spacing is 50kHz; if the mode is 4K mode, then the spacing is 25kHz. These values of subcarrier spacing (for upstream OFDMA channels) are defined in the Upstream OFDMA Parameters table in [PHYv4.0].

F.3.2 CM Downstream Interface Information Model

F.3.2.1 Overview

This section defines the configuration and status reporting requirements for the CM Downstream Interfaces. This information is contained in the [PHYv4.0] specification.

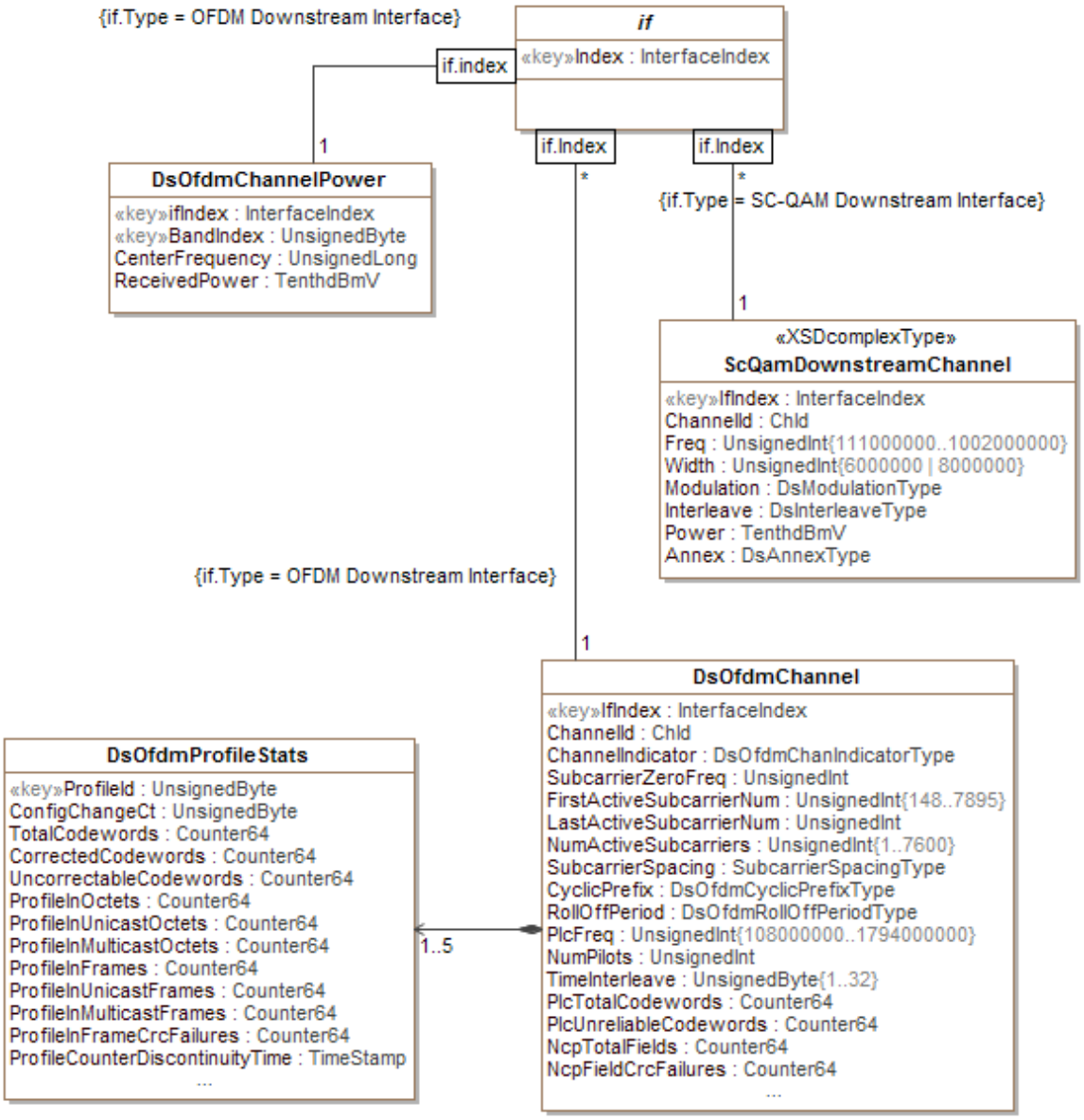


Figure 15 - CM Downstream Information Model

F.3.2.2 Data Type Definitions

Table 141 and the subsections which follow enumerate the CM downstream interface data types.

Table 141 - CM Downstream Parameter Data Types

Data Type Name	Base Type	Permitted Values	Reference
DsOfdmChanIndicatorType	UnsignedByte		
DsOfdmCyclicPrefixType	UnsignedShort	(192 | 256 | 512 | 768 | 1024)	[PHYv4.0]
DsOfdmRollOffPeriodType	UnsignedShort	(0 | 64 | 128 | 192 | 256)	[PHYv4.0]

Data Type Name	Base Type	Permitted Values	Reference
DsOfdmModulationType	Enum	other(1) zeroValued(2) qpsk(3) qam16(4) qam64(5) qam128(6) qam256(7) qam512(8) qam1024(9) qam2048(10) qam4096(11) qam8192 (12) qam16384 (13)	[PHYv4.0]

F.3.2.2.1 DsOfdmChanIndicatorType

This data type is defined to specify the channel indicator type for the downstream channel. The permitted values are 'other', 'primary', 'backupPrimary' and 'nonPrimary'.

F.3.2.2.2 DsOfdmCyclicPrefixType

This data type is defined to specify the five possible values for the length of cyclic prefix. The cyclic prefix (in Îźs) are converted into samples using the sample rate of 204.8 Msamples/s and is an integer multiple of: 1/64 * 20 µs. The possible values come from the Downstream OFDM Parameters table in [PHYv4.0].

F.3.2.2.3 DsOfdmRollOffPeriodType

This data type is defined to specify the five possible values for the windowing roll-off period. The Roll-Off Period is reported in 'number of samples'. The possible values come from the Downstream OFDM Parameters table in [PHYv4.0].

F.3.2.2.4 DsOfdmModulationType

This data type is defined to specify the modulation types supported by the CM demodulator. The values are defined in the Modulation Formats section of [PHYv4.0].

F.3.2.3 CM Downstream Interface Status Object Definitions

F.3.2.3.1 ScQamDownstreamChannel

This object reports the channel configuration of an SC-QAM downstream channel.

References: [RFC 4546], docsIfDownstreamChannelTable

Table 142 - ScQamDownstreamChannel Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
ChannelId	ChId	R/O	0..255		
Freq	UnsignedInt	R/O	111000000..1002000000	Hz	
Width	UnsignedInt	R/O	6000000 | 8000000	Hz	
Modulation	Enum	R/O	unknown(1) other(2) qam64(3) qam256(4)		

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Interleave	Enum	R/O	unknown(1) other(2) taps8Increment16(3) taps16Increment8(4) taps32Increment4(5) taps64Increment2(6) taps128Increment1(7) taps12increment17(8)		
Power	TenthdBmV	R/O		dBmV	
Annex	Enum	R/O	unknown(1) other(2) annexA(3) annexB(4) annexC(5)		

Table 143 - ScQamDownstreamChannel Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
if	Association	1	0	if.Type=Downstream Interface

F.3.2.3.1.1 IfIndex

This attribute is the interface index of the downstream interface and is a key to provide an index into the table.

F.3.2.3.1.2 ChannelId

This attribute is the Downstream Channel Identifier. This is an 8-bit identifier that distinguishes a Downstream Channel within a MAC Domain.

The Cable Modem Termination System identification of the downstream channel within this particular MAC interface. If the interface is down, the object returns the most current value. If the downstream channel ID is unknown, this object returns a value of 0.

Reference: [RFC 4546] docsIfDownChannelId

F.3.2.3.1.3 Freq

This attribute is the center of the downstream frequency associated with this channel. This object will return the current tuner frequency.

Reference: [RFC 4546] docsIfDownChannelFrequency

F.3.2.3.1.4 Width

This attribute is bandwidth of this downstream channel.

Reference: [RFC 4546] docsIfDownChannelWidth

F.3.2.3.1.5 Modulation

This attribute is the modulation of the channel.

Reference: [RFC 4546] docsIfDownChannelModulation

F.3.2.3.1.6 Interleave

This attribute is the Forward Error Correction (FEC) interleaving used for this downstream channel.

Reference: [RFC 4546] docsIfDownChannelInterleave

F.3.2.3.1.7 Power

This attribute is the received power level. If the interface is down, this object either returns the most recent value or the value of 0.

Reference: [RFC 4546] docsIfDownChannelPower

F.3.2.3.1.8 Annex

This attribute returns the Annex used by this channel. This value indicates the conformance of the implementation to important regional cable standards.

- annexA: Annex A from ITU-T J.83 is used. (equivalent to EN 300 429)
- annexB: Annex B from ITU-T J.83 is used.
- annexC: Annex C from ITU-T J.83 is used.

Reference: [RFC 4546] docsIfDownChannelAnnex

F.3.2.3.2 DsOfdmChannel

This object reports the configuration and statistics for a downstream OFDM channel.

The downstream OFDM channel boundary frequency limits depend on the mode of operation to which DOCSIS equipment is configured to operate. Refer to [PHYv3.1] *Downstream CM Spectrum* section, for the downstream OFDM boundary frequency limits when equipment is configured to be compliant with the DOCSIS 3.1 and (non-FDX and non-FDD extended spectrum) DOCSIS 4.0 frequency plans. Refer to [PHYv4.0] *Downstream FDX CM Spectrum* section for the downstream OFDM boundary frequency limits when equipment is configured to be compliant with FDX mode. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan for FDD Operation* section for the downstream OFDM boundary frequency limits when the equipment is configured to be compliant with FDD mode.

Table 144 - DsOfdmChannel Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
ChannelId	ChId	R/O			
ChannelIndicator	DsOfdmChanIndicatorType	R/O			
SubcarrierZeroFreq	UnsignedInt	R/O		Hz	
FirstActiveSubcarrierNum	UnsignedInt	R/O			
LastActiveSubcarrierNum	UnsignedInt	R/O			
NumActiveSubcarriers	UnsignedInt	R/O			
SubcarrierSpacing	SubcarrierSpacingType	R/O		kHz	
CyclicPrefix	DsOfdmCyclicPrefixType	R/O		samples	
RollOffPeriod	DsOfdmRollOffPeriodType	R/O		samples	
PlcFreq	UnsignedInt	R/O		Hz	
NumPilots	UnsignedInt	R/O			
TimeInterleaverDepth	UnsignedByte	R/O		symbols	
PlcTotalCodewords	Counter64	R/O			
PlcUnreliableCodewords	Counter64	R/O			
NcpTotalFields	Counter64	R/O			
NcpFieldCrcFailures	Counter64	R/O			

Table 145 - DsOfdmChannel Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
if	Association	1		if.Type=Downstream Interface
DsOfdmProfileStats	Directed Composition		1...5	

F.3.2.3.2.1 ifIndex

This attribute is the interface index of the downstream interface and is a key to provide an index into the table.

F.3.2.3.2.2 ChannelId

The CMTS identification of the downstream channel within this particular MAC interface. If the downstream channel Id is unknown, then this attribute returns a value of 0.

F.3.2.3.2.3 ChannelIndicator

This attribute is used to identify the OFDM downstream channel as primary, backup primary or non-primary. A value of 1 indicates that OFDM channel is assigned to be the CM's primary downstream channel. A value greater than 1 indicates that the OFDM channel is assigned to be the CM's backup primary downstream channel. A value of 0 indicates the OFDM channel is not assigned to be a CM's primary or backup primary downstream channel.

F.3.2.3.2.4 SubcarrierZeroFreq

This attribute specifies the center frequency of subcarrier 0 of the OFDM channel. This is the frequency of subcarrier X(0) in the definition of the Discrete Fourier Transform.

F.3.2.3.2.5 FirstActiveSubcarrierNum

This attribute corresponds to the number of the first non-excluded subcarrier.

F.3.2.3.2.6 LastActiveSubcarrierNum

This attribute corresponds to the number of the last non-excluded subcarrier.

F.3.2.3.2.7 NumActiveSubcarriers

This attribute represents the number of active data subcarriers within the OFDM downstream channel (i.e., this exclude subcarriers for continuous pilots and the PLC). For 4K FFT mode, the maximum number of subcarriers including continuous pilots and the PLC cannot exceed 3800, and for 8K FFT mode, the maximum number of active subcarriers including continuous pilots and the PLC cannot be greater than 7600. However, there are a minimum of 56 continuous pilots in a 192 MHz channel that has no exclusions, and the size of the PLC is 8 subcarriers for 4K FFT mode and 16 subcarriers for 8K FFT mode. Therefore, the maximum value of NumActiveSubcarriers is 3736 (or 3800 - 56 - 8) for 4K FFT mode and 7528 (or 7600 - 56 - 16) for 8K FFT mode.

F.3.2.3.2.8 SubcarrierSpacing

This attribute defines the subcarrier spacing associated with a particular FFT mode configured on the OFDM downstream channel. If it is 4K mode, then the subcarrier spacing is 50kHz. If it is 8K mode, then the subcarrier spacing is 25kHz.

F.3.2.3.2.9 CyclicPrefix

Cyclic prefix enables the receiver to overcome the effects of inter-symbol-interference and intercarrier-interference caused by micro-reflections in the channel. There are five possible values for the length of the CP and the choice depends on the delay spread of the channel - a longer delay spread requires a longer cyclic prefix.

Reference: [PHYv4.0] Cyclic Prefix Values

F.3.2.3.2.10 RollOffPeriod

Roll off period maximizes channel capacity by sharpening the edges of the spectrum of the OFDM signal. For windowing purposes another segment at the start of the IDFT output is appended to the end of the IDFT output -the roll-off postfix (RP). There are five possible values for the (RP), and the choice depends on the bandwidth of the channel and the number of exclusion bands within the channel. A larger RP provides sharper edges in the spectrum

of the OFDM signal; however, there is a time vs. frequency trade-off. Larger RP values reduce the efficiency of transmission in the time domain, but because the spectral edges are sharper, more useful subcarriers appear in the frequency domain. There is an optimum value for the RP that maximizes capacity for a given bandwidth and/or exclusion band scenario.

Reference: [PHYv4.0] Roll-off Period Values

F.3.2.3.2.11 PlcFreq

This attribute defines the location of the PHY Link Channel (PLC). It is the center frequency of the lowest subcarrier of the 6 MHz encompassed spectrum containing the PLC at its center.

F.3.2.3.2.12 NumPilots

This attribute indicates the number of continuous pilots configured for the downstream channel.

F.3.2.3.2.13 TimeInterleaverDepth

The attribute defines the depth of Time interleaving used for this downstream channel as received in the OCD message.

F.3.2.3.2.14 PlcTotalCodewords

This attribute represents the total number of PLC codewords received by the CM.

F.3.2.3.2.15 PlcUnreliableCodewords

This attribute represents the total number of PLC codewords which failed post-decoding LDPC syndrome check.

F.3.2.3.2.16 NcpTotalFields

This attribute represents the total number of NCP fields received by the CM.

F.3.2.3.2.17 NcpFieldCrcFailures

This attribute represents the total number of NCP fields received by the CM which failed the CRC check.

F.3.2.3.3 DsOfdmProfileStats

This CM object provides usage statistics for a modulation profile assigned to an OFDM downstream channel. A row entry is created when a profile is assigned. The row entry is deleted when a profile id becomes unassigned. The counts in this table are only of data on data profiles that is intended for this CM.

Table 146 - DsOfdmProfileStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
ProfileId	UnsignedByte	R/O	1..13|255		
ConfigChangeCt	UnsignedByte	R/O			
TotalCodewords	Counter64	R/O			
CorrectedCodewords	Counter64	R/O			
UncorrectableCodewords	Counter64	R/O			
ProfileInOctets	Counter64	R/O			
ProfileInUnicastOctets	Counter64	R/O			
ProfileInMulticastOctets	Counter64	R/O			
ProfileInFrames	Counter64	R/O			
ProfileInUnicastFrames	Counter64	R/O			
ProfileInMulticastFrames	Counter64	R/O			
ProfileInFrameCrcFailures	Counter64	R/O			
ProfileCounterDiscontinuityTime	TimeStamp	R/O			

F.3.2.3.3.1 ProfileId

This attribute is the unique identifier of the downstream profile associated with the OFDM downstream channel. It is a key defined to provide an index into the table. The Profile ID for Next Codeword Pointer (NCP) Profiles is 255 [MULPIv4.0].

F.3.2.3.3.2 ConfigChangeCt

This attribute contains the value of the Configuration Change Count field in the Downstream Profile Descriptor (DPD) MAC Management Message corresponding to this profile.

F.3.2.3.3.3 TotalCodewords

This attribute defines the total number of codewords (including full-length and shortened) measured on this profile.

F.3.2.3.3.4 CorrectedCodewords

This attribute defines the number of codewords measured on this profile that failed pre-decoding LDPC syndrome check and passed BCH decoding.

F.3.2.3.3.5 UncorrectableCodewords

This attribute defines the number of codewords measured on this profile that failed BCH decoding for data profile and post-decoding LDPC syndrome check for NCP profile.

F.3.2.3.3.6 ProfileInOctets

This attribute is the count of MAC-layer octets received by the CM on this profile. This value is the size of all unicast, multicast or broadcast frames (including all MAC-layer framing) delivered from the PHY layer to the MAC layer - this includes user data, DOCSIS MAC Management Messages, etc.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.7 ProfileInUnicastOctets

This attribute is the count of MAC-layer unicast octets received by the CM on this profile. This value is the size of all unicast frames (including all MAC-layer framing) delivered from the PHY layer to the MAC layer - this includes user data, DOCSIS MAC Management Messages, etc.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.8 ProfileInMulticastOctets

This attribute is the count of MAC-layer multicast and broadcast octets received by the CM on this profile. This value is the size of all frames (including all MAC-layer framing) delivered from the PHY layer to the MAC layer and addressed to a multicast MAC address - this includes user data, DOCSIS MAC Management Messages, etc.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.9 ProfileInFrames

This attribute is the count of frames received by the CM on this profile. This value is the count of all unicast, multicast or broadcast frames delivered from the PHY layer to the MAC layer - this includes user data, DOCSIS MAC Management Messages, etc.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.10 ProfileInUnicastFrames

This attribute is the count of unicast frames received by the CM on this profile. This value is the count of all frames delivered from the PHY layer to the MAC layer and addressed to a unicast MAC address - this includes user data, DOCSIS MAC Management Messages, etc.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.11 ProfileInMulticastFrames

This attribute is the count of multicast frames received by the CM on this profile. This value is the count of all frames delivered from the PHY layer to the MAC layer and addressed to a multicast MAC address - this includes user data, DOCSIS MAC Management Messages, etc., but excludes frames sent to a broadcast address.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.12 ProfileInFrameCrcFailures

This attribute is the count of frames received by the CM on this profile that failed the MAC frame CRC check.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.3.13 ProfileCounterDiscontinuityTime

This attribute is the value of sysUpTime on the most recent occasion at which any one or more of this entry's counters suffered a discontinuity (e.g., counter rollover, other vendor-specific event). If no such discontinuities have occurred since the last re-initialization of the local management subsystem, then this attribute contains a zero value.

The CM reports zero for this attribute for Next Codeword Pointer profile.

F.3.2.3.4 DsOfdmChannelPower

This object provides the attributes to measure the channel power for a 6 MHz wide band at the F connector input of the CM.

Table 147 - DsOfdmChannelPower Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
ifIndex	InterfaceIndex	Key			
BandIndex	UnsignedByte	Key	0..33		
CenterFrequency	UnsignedLong	R/O	111000000.. 1791000000	Hz	0
ReceivedPower	TenthdBmV	R/O		dBmV	

Table 148 - DsOfdmChannelPower Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
if	Association	1		if.Type=Downstream Interface

F.3.2.3.4.1 ifIndex

This attribute is the interface index of an OFDM downstream interface in the current CM receive channel set and is a key to provide an index into the table.

F.3.2.3.4.2 BandIndex

This attribute is a unique index used by the CM to identify each of the 6 MHz bands of a given OFDM downstream channel (from the lowest 6 MHz band of the Modulated Spectrum to the highest 6 MHz band of the Modulated

Spectrum). The CM MUST assign indices in frequency order from the OFDM channel's lowest to highest 6 MHz frequency band for each of the 6 MHz bands of the channel, using an index of 1 to represent the lowest frequency band of the Modulated Spectrum. Thus, an index of 33 represents the highest possible 6 MHz frequency band of the Modulated Spectrum of a DOCSIS 4.0 OFDM channel. If there are interior Exclusion Bands resulting in 6 MHz bands which contain no Active Subcarriers, then the indices corresponding to those bands will be skipped and the power for those bands will not be reported. The CM MUST also provide the power of the PLC channel and utilize an index value of 0 to represent the PLC channel in this table. If the placement of the OFDM channel is such that the center frequency of the lowest or highest active subcarrier is placed on a 1 MHz grid and exactly at the edge of two 6 MHz bands, then the band which contains only the spectral edge of that subcarrier will not be included in the list of 6 MHz bands even though by definition, that band would contain a minute portion of the Modulated Spectrum. This logic also applies to any 6 MHz bands which are skipped due to interior Exclusion Bands.

F.3.2.3.4.3 CenterFrequency

This attribute corresponds to the center frequency of the 6 MHz band where the CM measured the average channel power. The 6 MHz measurement band is defined as any 6 MHz band with a center frequency of 111 + 6 (n-1) MHz for n = 1,2...281 (i.e., 111,117,...1791 MHz).

The CM MUST provide the center frequency for the 6 MHz channel, other than the one encompassing the PLC channel, per the following formula:

centerfreq = 111 + 6(n-1)

such that (centerfreq - 111) / 6, is a whole number.

If the CM is reporting the frequency of the PLC, using a BandIndex of 0, the CM MUST provide the center frequency of the lowest subcarrier of the 6 MHz encompassed spectrum containing the PLC at its center.

The downstream OFDM channel center frequency range depends on the mode of operation to which DOCSIS equipment is configured to operate. Refer to [PHYv3.1] *Downstream CM Spectrum* section, for the downstream OFDM boundary frequency limits when equipment is configured to be compliant with the DOCSIS 3.1 and (non-FDX and non-FDD extended spectrum) DOCSIS 4.0 frequency plans. Refer to [PHYv4.0] *Downstream FDX CM Spectrum* section for the downstream OFDM boundary frequency limits when equipment is configured to be compliant with FDX mode. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan for FDD Operation* section for the downstream OFDM boundary frequency limits when the equipment is configured to be compliant with FDD mode.

F.3.2.3.4.4 ReceivedPower

This attribute provides an estimate of the average power measured at the F connector input of the CM in the receive downstream channel set for any 6 MHz bandwidth with the Center Frequency of 111 + 6(n-1) MHz for n =1,2,...185 (i.e., 111,117,...1215 MHz).

If the Center Frequency is 0, then this attribute provides an estimate of the average power measured at the F connector input of the CM for a 6 MHz encompassed spectrum containing the DOCSIS 4.0 PLC at its center.

F.3.3 CM Upstream Interface Information Model

F.3.3.1 Overview

This section defines the configuration and status reporting requirements for the CM Upstream Interface. This information is contained in the [PHYv4.0] specification.

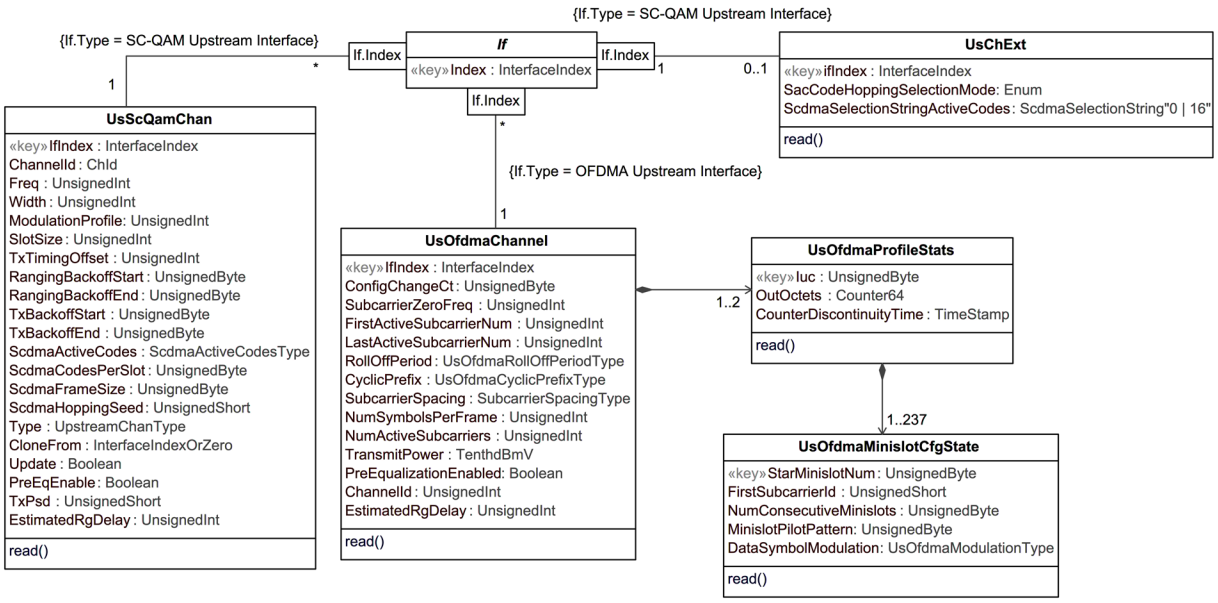


Figure 16 - CM Upstream Information Model

F.3.3.2 Data Type Definitions

The following table and subsections define the CM upstream interface data types.

Table 149 - Data Types

Data Type Name	Base Type	Permitted Values	Reference
UsCyclicPrefixType	Unsigned Short	(96 | 128 | 160 | 192 | 224 | 256 | 288 | 320 | 384 | 512 | 640)	[PHYv4.0]
UsOfdmaRollOffPeriodType	UnsignedByte	(0 | 32 | 64 | 96 | 128 | 160 | 192 | 224)	[PHYv4.0]
UsOfdmaModulationType	Enum	other(1) zeroValued(2) bpsk(3) qpsk(4) qam8(5) qam16(6) qam32(7) qam64(8) qam128(9) qam256(10) qam512(11) qam1024(12) qam2048(13) qam4096(14)	[PHYv4.0]
UsOfdmaSubcarrierperMinislotType	UnsignedByte	(8 | 16)	[PHYv4.0]

F.3.3.2.1 UsOfdmaCyclicPrefixType

This data type is defined to specify the eleven possible values for the length of cyclic prefix. The cyclic prefix (in µs) are converted into samples using the sample rate of 102.4 Msamples/s and is an integer multiple of: 1/64 * 20 µs. The possible values come from the Upstream OFDMA Parameters table in [PHYv4.0].

F.3.3.2.2 UsOfdmaSubcarrierperMinislotType

This data type defines the number of subcarriers per minislot. For 2K mode, its value is 8 and for 4K mode, it is 16. The possible values are defined in the Minislot Parameters table of [PHYv3.1].

F.3.3.2.3 UsOfdmaRollOffPeriodType

This data type is defined to specify the eight possible values for the windowing roll-off period. The Roll-Off Period is given in number of samples using the sample rate of 102.4 Msamples/s. The possible values come from the Upstream OFDMA Parameters table in [PHYv4.0].

F.3.3.2.4 UsOfdmaModulationType

This data type is defined to specify the modulation types supported by the CM modulator. The values are defined in the Modulation Formats section of [PHYv4.0].

F.3.3.3 Object Definitions

F.3.3.3.1 UsScQamChan

The UsScQamChan object provides the configuration state of an upstream SC-QAM channel.

Table 150 - UsScQamChan Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
IfIndex	InterfaceIndex	Key			
ChannelId	ChId	R/O	0..255		
Freq	UnsignedInt	R/O	5800000..84200000	Hz	
Width	UnsignedInt	R/O	1600000 | 3200000 | 6400000	Hz	
ModulationProfile	UnsignedInt	R/O			
SlotSize	UnsignedInt	R/O		ticks	
TxTimingOffset	UnsignedInt	R/O			
RangingBackoffStart	UnsignedByte	R/O	0..16		
RangingBackoffEnd	UnsignedByte	R/O	0..16		
TxBackoffStart	UnsignedByte	R/O	0..16		
TxBackoffEnd	UnsignedByte	R/O	0..16		
ScdmaActiveCodes	UnsignedInt	R/O	0 | 64..66 |68..70 |72 |74..78 |80..82 |84..88 |90..96 |98..100 |102 |104..106 |108 |110..112 |114..126 |128		
ScdmaCodesPerSlot	UnsignedByte	R/O	0 | 2..32		
ScdmaFrameSize	UnsignedByte	R/O	0 | 2..32		
ScdmaHoppingSeed	UnsignedShort	R/O	0..32767		
Type	Enum	R/O	unknown(0) tdma(1) atdma(2) scdma(3) tdmaAndAtdma(4)		
CloneFrom	InterfaceIndexOrZero	R/O			
Update	Boolean	R/O			
PreEqEnable	Boolean	R/O			
TxPsd	UnsignedShort	R/O		quarter dBmV	
EstimatedRgDelay	UnsignedInt	R/O		microseconds	

Table 151 - UsScQamChan Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
if	Association	1	0	if.Type=SC-QAM Upstream Interface

F.3.3.3.1.1 IfIndex

This attribute is the interface index of the upstream interface and is a key to provide an index into the table.

F.3.3.3.1.2 ChannelId

This attribute is the Upstream Channel Identifier.

Reference: [RFC 4546] docsIfUpChannelId

F.3.3.3.1.3 Freq

This attribute is the center of the frequency band associated with this upstream interface. This object returns 0 if the frequency is undefined or unknown.

Reference: [RFC 4546] docsIfUpChannelFrequency

F.3.3.3.1.4 Width

This attribute is the bandwidth of this upstream interface. This object returns 0 if the interface width is undefined or unknown.

Reference: [RFC 4546] docsIfUpChannelWidth

F.3.3.3.1.5 ModulationProfile

This attribute is the modulation profile for the upstream channel.

Reference: [RFC 4546] docsIfUpChannelModulationProfile

F.3.3.3.1.6 SlotSize

This attribute is the slot size for the upstream channel. It is applicable to TDMA and ATDMA channel types only. The number of 6.25 microsecond ticks in each upstream minislot. This object returns zero if the value is undefined or unknown or in case of an SCDMA channel.

Reference: [RFC 4546] docsIfUpChannelSlotSize

F.3.3.3.1.7 TxTimingOffset

This attribute is the measure of the current round-trip time obtained from the ranging offset (initial ranging offset + ranging offset adjustments). Units are one 64th fraction of 6.25 microseconds."

Reference: [RFC 4546] docsIfUpChannelTxTimingOffset

F.3.3.3.1.8 RangingBackoffStart

This attribute is the initial random backoff window the CM will use when retrying Ranging Requests. Expressed as a power of 2. A value of 16 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used.

Reference: [RFC 4546] docsIfUpChannelRangingBackoffStart

F.3.3.3.1.9 RangingBackoffEnd

This attribute is the final random backoff window the CM will use when retrying Ranging Requests. Expressed as a power of 2. A value of 16 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used.

Reference: [RFC 4546] docsIfUpChannelRangingBackoffEnd

F.3.3.3.1.10 TxBackoffStart

This attribute is the initial random backoff window the CM will use when retrying transmissions. Expressed as a power of 2. A value of 16 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used.

Reference: [RFC 4546] docsIfUpChannelTxBackoffStart

F.3.3.3.1.11 TxBackoffEnd

This attribute is the final random backoff window the CM will use when retrying transmissions. Expressed as a power of 2. A value of 16 at the CMTS indicates that a proprietary adaptive retry mechanism is to be used.

Reference: [RFC 4546] docsIfUpChannelTxBackoffEnd

F.3.3.3.1.12 ScdmaActiveCodes

This attribute is the SCDMA Active Codes. It is applicable for SCDMA channel types only. This object returns the number of active codes. It returns zero for non-SCDMA channel types. Note that legal values for ScdmaActiveCodes from 64..128 MUST be non-prime.

Reference: [RFC 4546] docsIfUpChannelScdmaActiveCodes

F.3.3.3.1.13 ScdmaCodesPerSlot

This attribute is the number of SCDMA codes per minislot. It is applicable for SCDMA channel types only.

It returns zero if the value is undefined or unknown or in case of a TDMA or ATDMA channel.

Reference: [RFC 4546] docsIfUpChannelScdmaCodesPerSlot

F.3.3.3.1.14 ScdmaFrameSize

This attribute is the SCDMA frame size in units of spreading intervals. It is applicable for SCDMA channel types only. This value returns zero for non-SCDMA Profiles.

Reference: [RFC 4546] docsIfUpChannelScdmaFrameSize

F.3.3.3.1.15 ScdmaHoppingSeed

This attribute is the 15-bit seed used for code hopping sequence initialization. It is applicable for SCDMA channel types only. This object returns zero for non-SCDMA channel types.

Reference: [RFC 4546] docsIfUpChannelScdmaHoppingSeed

F.3.3.3.1.16 Type

This attribute reflects the Upstream channel type.

Reference: [RFC 4546] docsIfUpChannelType

F.3.3.3.1.17 CloneFrom

This attribute is meaningless on a Cable Modem.

Reference: [RFC 4546] docsIfUpChannelCloneFrom

F.3.3.3.1.18 Update

This attribute always returns 'false' on a Cable Modem.

Reference: [RFC 4546] docsIfUpChannelUpdate

F.3.3.3.1.19 PreEqEnable

This attribute reflects the status of pre-equalization as represented in the RNG-RSP. Pre-equalization is considered enabled at the CM if a RNG-RSP with pre-equalization data has been received at least once since the last mac reinitialization.

Reference: [RFC 4546] docsIfUpChannelPreEqEnable

F.3.3.3.1.20 TxPsd

This attribute represents $P_{1.6r\_n}$, the power spectral density in 1.6 MHz, for the associated SC-QAM upstream channel.

F.3.3.3.1.21 EstimatedRgDelay

This attribute reports the estimate of the Request-Grant delay for this upstream channel, as calculated in the calcAllowedAQ function. The value is used by the CM for setting the AllowedAqBytes value for any Low Latency Service Flow that uses this upstream channel. If a particular upstream channel is not used by any Low Latency Service Flow, the CM MAY report 0 for the EstimatedRgDelay attribute.

Reference: [MULPIv3.1] estimatedRgDelay parameter calculated via the 'calcAllowedAQ()' function in Annex O.1 AQM Utility Functions

F.3.3.3.2 UsChExt

This object defines management extensions for upstream channels, in particular SCDMA parameters.

Table 152 - UsChExt Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	key	InterfaceIndex of MAC Domain interface		
SacCodeHoppingSelectionMode	Enum	R/O	none(0) sac1NoCodeHopping(1) sac1CodeHoppingMode1(2) sac2CodeHoppingMode2(3) sac2NoCodeHopping(4)		
ScdmaSelectionStringActiveCodes	ScdmaSelectionString	R/O			

F.3.3.3.2.1 IfIndex

This key represents the interface index of the logical upstream channel to which this instance applies.

F.3.3.3.2.2 SacCodeHoppingSelectionMode

This attribute indicates the selection mode for active codes and code hopping.

- 'none'
  Non-SCDMA channel
- 'sac1NoCodeHopping'
  Selectable active codes mode 1 and code hopping disabled
- 'sac1CodeHoppingMode1'
  Selectable active codes mode 1 and code hopping mode 1
- 'sac2CodeHoppingMode2'
  Selectable active codes mode 2 and code hopping mode 2
- 'sac2NoCodeHopping'
  Selectable active codes mode 2 and code hopping disabled

References: Minislot Numbering Parameters in Timing and Synchronization section.

F.3.3.3.2.3 ScdmaSelectionStringActiveCodes

This attribute represents the active codes of the upstream channel and it is applicable only when SacCodeHoppingSelectionMode is 'sac2CodeHoppingMode2'.

References: [MULPIv4.0] Minislot Numbering Parameters in Timing and Synchronization section.

F.3.3.3.3 UsOfdmaChannel

The UsOfdmaChannel object reports the configuration properties of an upstream OFDMA channel.

The upstream OFDMA channel band edge limits depend on the mode of operation to which DOCSIS equipment is configured to operate. Refer to [PHYv3.1] *Upstream CM Spectrum* section, for the upstream OFDMA channel boundary frequency limits when equipment is configured to be compliant with the DOCSIS 3.1 and (non-FDX and

non-FDD extended spectrum) DOCSIS 4.0 frequency plans. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan* section for the upstream OFDMA channel boundary frequency limits when equipment is configured to be compliant with FDX mode. Refer to [PHYv4.0] *Upstream and Downstream Frequency Plan for FDD Operation* section for the upstream OFDMA channel boundary frequency limits when the equipment is configured to be compliant with FDD mode.

Table 153 - UsOfdmaChannel Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Ifindex	InterfaceIndex	Key			
ConfigChangeCt	UnsignedByte	R/O			
SubcarrierZeroFreq	UnsignedInt	R/O		Hz	
FirstActiveSubcarrierNum	UnsignedInt	R/O			
LastActiveSubcarrierNum	UnsignedInt	R/O			
RollOffPeriod	UsOfdmaRollOffPeriodType	R/O		Number of samples	
CyclicPrefix	UsOfdmaCyclicPrefixType	R/O		Number of samples	
SubcarrierSpacing	SubcarrierSpacingType	R/O		Hz	
NumSymbolsPerFrame	UnsignedInt	R/O			
NumActiveSubcarriers	UnsignedInt	R/O			
TransmitPower	UnsignedInt	R/O		Quarter dBmV	
PreEqualizationEnabled	Boolean	R/O			
ChannelId	UnsignedInt	R/O			
EstimatedRgDelay	UnsignedInt	R/O		microseconds	

Table 154 - UsOfdmaChannel Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
if	Association	1		if.Type=Upstream Interface
UsOfdmaProfileStats	Directed Composition		1..2	

F.3.3.3.3.1 ifIndex

This attribute is the interface index of the upstream interface and is a key to provide an index into the table.

F.3.3.3.3.2 ConfigChangeCt

This attribute contains the value of the Configuration Change Count field in the Upstream Channel Descriptor (UCD) MAC Management Message corresponding to this upstream channel.

F.3.3.3.3.3 SubcarrierZeroFreq

This attribute specifies the center frequency of subcarrier 0 of the OFDMA channel. Note that since subcarrier 0 is always excluded, it will actually be below the allowed upstream spectrum band.

F.3.3.3.3.4 FirstActiveSubcarrierNum

This attribute corresponds to the index of the first non-excluded subcarrier.

F.3.3.3.3.5 LastActiveSubcarrierNum

This attribute corresponds to the index of the last non-excluded subcarrier.

F.3.3.3.3.6 RollOffPeriod

Windowing is applied in order to maximize channel capacity by sharpening the edges of the spectrum of the OFDMA signal. Windowing is applied in the time domain by tapering (or rolling off) the edges using a raised cosine function. There are eight possible values of roll-off prefix. The Roll-Off Period is given in the number of samples

using the sample rate of 102.4 Msamples/s. The configuration where Roll-off prefix value is greater than or equal to cyclic prefix value is considered invalid.

F.3.3.3.3.7 CyclicPrefix

Cyclic prefix is added in order to enable the receiver to overcome the effects of inter-symbol interference (ISI) and inter-carrier interference caused by microreflections in the channel. The cyclic prefix (in µs) is converted into samples using the sample rate of 102.4 Msamples/s. There are eleven possible values for the length of the CP and the choice depends on the delay spread of the channel - a longer delay spread requires a longer cyclic prefix.

F.3.3.3.3.8 SubcarrierSpacing

This attribute defines the subcarrier spacing associated with a particular FFT mode configured on the OFDMA upstream channel. If it is 2K mode, then the subcarrier spacing is 50kHz. If it is 4K mode, then the subcarrier spacing is 25kHz.

F.3.3.3.3.9 NumSymbolsPerFrame

This attribute defines the number of symbol periods per frame. For channel bandwidth greater than 72MHz, the maximum number of symbol periods per frame is 18 for 2K mode and 9 for 4K mode. For channel bandwidth less than 72 MHz but greater than 48MHz, the maximum number of symbols per frame is 24 for 2K mode and 12 for 4K mode. For channel bandwidth less than 48MHz, the maximum number of symbol periods is 36 for 2K mode and 18 for 4K mode. The minimum number of symbol periods per frame is 6 for both the FFT modes and is independent of the channel bandwidth.

F.3.3.3.3.10 NumActiveSubCarriers

This attribute defines the number of active subcarriers within the OFDMA upstream channel.

F.3.3.3.3.11 TransmitPower

This attribute represents the operational transmit power for the associated OFDMA upstream channel. The CM reports its Target Power, $P_{1.6r\_n}$ as described in [PHYv4.0].

F.3.3.3.3.12 PreEqualizationEnabled

This attribute defines whether pre-equalization is enabled on the associated OFDMA upstream channel.

F.3.3.3.3.13 ChannelId

This attribute us the upstream OFDMA channel identifier. This is an 8-bit identifier that uniquely identifies an OFDMA upstream channel within a MAC domain.

If the interface is down, the object returns the most current value. If the upstream channel ID is unknown, this object returns a value of 0.

F.3.3.3.3.14 EstimatedRgDelay

This attribute reports the estimate of the Request-Grant delay for this upstream channel, as calculated in the calcAllowedAQ function. The value is used by the CM for setting the AllowedAqBytes value for any Low Latency Service Flow that uses this upstream channel. If a particular upstream channel is not used by any Low Latency Service Flow, the CM MAY report 0 for the EstimatedRgDelay attribute.

Reference: [MULPIv3.1] estimatedRgDelay parameter calculated via the 'calcAllowedAQ()' function in Annex O.1 AQM Utility Functions

F.3.3.3.4 UsOfdmaProfileStats

This object provides usage statistics for an upstream OFDMA profile. The CM MUST create an instance of the UsOfdmaProfileStats object for each profile associated with the OFDMA upstream channel assigned to the cable modem.

Table 155 - UsOfdmaProfileStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Iuc	UnsignedByte	Key			
OutOctets	Counter64	R/O			
CounterDiscontinuityTime	TimeStamp	R/O			

Table 156 - UsOfdmaProfileStats Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
UsOfdmaMinislotCfgState	Directed Composition		0..237	

F.3.3.3.4.1 Iuc

This attribute is the unique identifier of the upstream profile associated with the OFDMA upstream channel (in the upstream direction the ProfileId is synonymous with the IUC). It is a key defined to provide an index into the table.

F.3.3.3.4.2 OutOctets

This attribute is the count of MAC-layer octets transmitted by the CM using this profile. This value is the size of all unicast, multicast or broadcast frames (including all MAC-layer framing) delivered from the MAC to the Phy - this includes user data, DOCSIS MAC Management Messages, etc.

Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ProfileCounterDiscontinuityTime.

F.3.3.3.4.3 CounterDiscontinuityTime

This attribute is the value of sysUpTime on the most recent occasion at which any one or more of this entry's counters suffered a discontinuity. If no such discontinuities have occurred since the last re-initialization of the local management subsystem, then this attribute contains a zero value.

F.3.3.3.5 UsOfdmaMinislotCfgState

This CM object reports minislot configuration as received in the UCD message for a particular OFDMA profile.

Table 157 - UsOfdmaMinislotCfgState Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
StartMinislotNum	UnsignedByte	Key	1..237		
FirstSubcarrierId	UnsignedShort	R/O	1..4095		
NumConsecutiveMinislots	UnsignedShort	R/O	1..237		
MinislotPilotPattern	UnsignedByte	R/O	1..14		
DataSymbolModulation	UsOfdmaModulationType	R/O			

F.3.3.3.5.1 StartMinislotNum

This attribute corresponds to the unique identifier of the minislot received by the CM. It is a key defined to provide an index into the table.

F.3.3.3.5.2 FirstSubcarrierId

This attribute corresponds to the index of the first/starting subcarrier in this minislot.

F.3.3.3.5.3 NumConsecutiveMinislots

This attribute defines the number of continuous minislots which have the same bit loading, starting with the StartMinislotNum, defined in the associated upstream profile.

F.3.3.3.5.4 MinislotPilotPattern

This attribute defines the pilot pattern used for edge and body minislots. Pilots are used by the CMTS receiver to adapt to channel conditions and frequency offset. Pilot patterns differ by the number of pilots in a minislot, and by their arrangement within the minislot. For both 8 and 16 subcarriers minislot sizes, seven pilot patterns are defined.

F.3.3.3.5.5 DataSymbolModulation

This attribute defines the bit loading within the minislot.

F.3.4 FDX RBA Report Information Model

F.3.4.1 Overview

This section defines the status reporting requirements for the CM FDX RBA functionality.

Reference: [MULPIv4.0] Resource Block Assignment section.

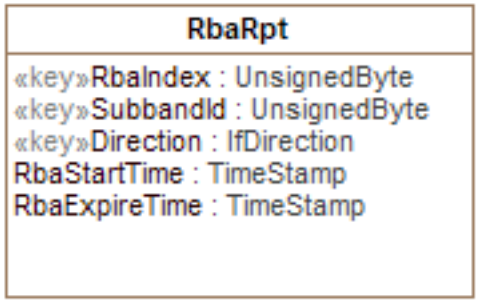


Figure 17 - CM Resource Block Assignment Report Information Model

F.3.4.2 Object Definitions

F.3.4.2.1 RbaRpt

This object reports the current FDX RBA last received by the cable modem. Instances are created when RBA Reporting is enabled. The CM will only report the RBA information that was received during the time period starting at the time reported by RbaRptCfg attribute SchedStartTime and ending at the time (SchedStartTime + *d*), where *d* is the value of RbaRptCfg attribute Duration.

The CM will not persist instances of this object across CM reboots. If RBA Reporting is enabled after Duration has expired, the CM will overwrite any previous instances of RbaRpt and create new instances.

Table 158 - RbaRpt Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default Value
Index	UnsignedShort	Key	1		
SubbandId	UnsignedByte	Key	1..3		
Direction	IfDirection	Key			
RbaStartTime	TimeStamp	R/O			
RbaExpireTime	TimeStamp	R/O			

F.3.4.2.1.1 Index

This attribute uniquely identifies each RBA received by the cable modem during the enabled duration. The only value supported is '1'. This provides a single snapshot when queried by the NMS.

F.3.4.2.1.2 SubbandId

This attribute reports the sub-band for which the RBA is to be scheduled.

Reference: [PHYv4.0] Full Duplex Channel Band Rules section.

F.3.4.2.1.3 Direction

This attribute indicates the data flow direction for each sub-band identified in the RBA.

F.3.4.2.1.4 RbaStartTime

This attribute reports the 32-bit DOCSIS timestamp defining when the Resource Block begins. The reported time could be prior to the current time.

F.3.4.2.1.5 RbaExpireTime

This attribute reports the 32-bit DOCSIS timestamp defining when the Resource Block assignment described in this RBA expires.

Annex G MAC and Upper Layer Protocols Interface (MULPI) Requirements (Normative)

G.1 Overview

This Annex defines management object extensions for Media Access Control (MAC) information, including DOCSIS interface configuration, RF Topology, Channel Bonding, QoS, and related extensions.

G.1.1 Cable Modem Service Groups (CM-SGs)

The HFC RF combining and splitting topology between a CMTS and Cable Modems results in distinct sets of CMs called Cable Modem Service Groups (CM-SGs) that are served by distinct combinations (i.e., non-overlapping subsets) of Downstream Channels and Upstream Channels. Because a MAC Domain defines a separate number space for many DOCSIS protocol elements (e.g., DSIDs, SAIDs, etc.), an operator should define separate MAC Domains that serve disjoint subsets of CM-SGs rather than a single MAC Domain for all CM-SGs.

G.1.2 Downstream Bonding Group (DBG)

A Downstream Bonding Group (DBG) is a set of Downstream Channels (DCs) on which the CMTS distributes packets. The CMTS enforces that all Downstream Channels of a DBG are contained within the same MAC Domain Downstream Service Group (MD-DS-SG). A CMTS permits configuration of a Downstream Channel as a member of multiple DBGs. A CMTS can restrict the assignment of Downstream Channels to DBGs based on vendor product implementation. For example, a CMTS product implementation may restrict the set of Downstream Channels that could be bonded to a given Bonded Channel Set to a subset of the downstream channels in the MAC Domain.

G.1.3 Upstream Bonding Group (UBG)

An Upstream Bonding Group (UBG) is a set of Upstream Channels (UCs) on which upstream data forwarding service may be provided to a single CM. All Upstream Channels in an Upstream Bonding Group need to be contained within the same MAC Domain Upstream Service Group (MD-US-SG). A CMTS permits configuration of an Upstream Channel as a member of multiple UBGs. A CMTS can restricts the assignment of Upstream Channels to UBGs based on vendor product implementation. For example, a CMTS product implementation could restrict the set of Upstream Channels that could be bonded to a subset of the downstream channels in the MAC Domain.

G.2 Object Definitions

This section defines the MULPI objects including the associated attributes.

G.2.1 Type Definitions

This section defines data types used in the object definitions for the MULPI information model.

Table 159 - Data Type Definitions

Data Type Name	Base Type	Permitted Values
AttrAggrRuleMask	HexBinary	SIZE (4)
AttributeMask	EnumBits	bonded(0) lowLatency(1) highAvailability(2)
BitRate	UnsignedInt	0..4294967295
ChannelList	HexBinary	SIZE (0..255)
ChId	UnsignedByte	0..255
ChSetId	UnsignedInt	0..4294967295
CpeInterfaceMaskType	See [CCAP-OSSIv4.0]	

Data Type Name	Base Type	Permitted Values
DataRateUnitType	Enum	bps(0), kbps(1), mbps(2), gbps(3)
Dsid	UnsignedInt	0..1048575
IfDirection	Enum	downstream (1) upstream (2)
NodeName	String	SIZE(0..64)
OfdmProfiles	EnumBits	profile0(0) profile1(1) profile2(2) profile3(3) profile4(4) profile5(5) profile6(6) profile7(7) profile8(8) profile9(9) profile10(10) profile11(11) profile12(12) profile13(13) profile14(14) profile15(15)
PrimaryDsIndicatorType	Enum	other (1) primaryDsChannel (2) backupPrimaryDs (3) notSpecified(4)
RcpId	HexBinary	SIZE (5)
ScdmaSelectionString	HexBinary	SIZE (0 | 16)
SchedulingType	Enum	undefined (1) bestEffort (2) nonRealTimePollingService (3) realTimePollingService (4) unsolicitedGrantServiceWithAD (5) unsolicitedGrantService (6) proactiveGrantService(7)

G.2.1.1 AttrAggrRuleMask

This data type represents a sequence of 32-bit positions that defines logical (e.g., AND, OR) operations to match against the channel list Provisioned Mask and Service Flow Required Mask bit positions when the CMTS is determining the service flow for assignment to a bonding group not configured by the management system.

References: [MULPIv4.0] Service Flow Assignment section.

G.2.1.2 AttributeMask

This data type consists of a sequence of 32-bit positions used to select the bonding group or the channel to which a service flow is assigned. DOCSIS defines three types of Attribute Masks for which this type applies: The Provisioned Attribute Mask that is configured to a Bonding Group or a single-channel, whereas the Required Attribute and the Forbidden Attribute Mask are part of the Service Flow QoS Parameter Set to be matched with the Provisioned Attribute Mask of CMTS-configured Bonding Groups or single-channels. DOCSIS reserves the assignment of the meaning of the first 8-bit positions (left to right) as follows:

Bit 0: 'bonding'

Bit 1: 'lowLatency'

Bit 2: 'highAvailability'

Bit positions 3-15 are reserved.

Bit positions 16-31 are freely assigned by operators to represent their own constraints on the channel(s) selection for a particular service flow.

References: [MULPIv4.0] Service Flow Assignment section

G.2.1.3 BitRate

This data type represents the rate of traffic. The units are specified by a multiplier attribute, DataRateUnitSetting, as bits per second (bps or bits/s), kilobits per second (kbps or kbit/s), megabits per second (Mbps or Mbit/s), or gigabits per second (Gbps or Gbit/s).

Kilobits per second is measured as 1,000 bits per second.

Megabits per second is measured as 1 million bits per second.

Gigabits per second is measured as 1 billion bits per second.

G.2.1.4 ChannelList

This data type represents a unique set of channel IDs in either the upstream or the downstream direction. Each octet represents a UCID or DCID depending on the direction of the channels within the list. The CMTS ensures that this combination of channels is unique per direction within the MAC Domain.

A query to retrieve the value of an attribute of this type, returns the set of channels in the channel list in ascending order of Channel Ids.

G.2.1.5 ChId

This data type is an 8-bit number that represents a provisioned Downstream Channel ID (DCID) or a provisioned Upstream Channel ID (UCID). A Channel Id is unique per direction within a MAC Domain. The value zero is reserved for use when the channel ID is unknown.

References: [MULPIv4.0] Upstream Channel Descriptor (UCD) section.

G.2.1.6 ChSetId

This data type is a CMTS-derived unique number within a MAC Domain used to reference a Channel Set within the CMTS. Values in the range of 1 to 255 define a single-channel Channel Set and correspond to either the Downstream Channel ID (DCID) or an Upstream Channel ID (UCID) of that channel. Values greater than 255 indicate a Channel Set consisting of two or more channels in the same direction within the MAC Domain. The value zero is reserved for use when the Channel Set is unknown.

References: [MULPIv4.0] Channel Bonding section.

G.2.1.7 CpeInterfaceMaskType

References: [CCAP-OSSIv4.0] CCAP Data Type Definitions section.

G.2.1.8 DataRateUnitType

This data type specifies the base unit for traffic rate parameters. The value of this data type allows for their interpretation in units of bps, kbps, Mbps, or Gbps. The enumeration starts from 0 to match corresponding DOCSIS protocol TLV values.

Kilobits per second is measured as 1,000 bits per second.

Megabits per second is measured as 1 million bits per second.

Gigabits per second is measured as 1 billion bits per second.

G.2.1.9 Dsid

This data type defines the 20-bit Downstream Service Identifier used by the CM for downstream resequencing, filtering, and forwarding. The value zero is reserved for use when the DSID is unknown or does not apply.

References: [MULPIv4.0] DSID Definition section.

G.2.1.10 IfDirection

Indicates a direction on an RF MAC interface. The value downstream(1) is from Cable Modem Termination System to Cable Modem. The value upstream(2) is from Cable Modem to Cable Modem Termination System.

Valid enumerations for the data type are:

- downstream(1)
- upstream(2)

Reference: [MULPIv4.0] Terms and Definitions section.

G.2.1.11 NodeName

This data type is a human readable string that represents the name of a fiber node. Internationalization is supported by conforming to the SNMP textual convention SnmpAdminString. The US-ASCII control characters (0x00 - 0x1F), the DEL character (0x7F), and the double-quote mark (0x22) are prohibited within the syntax of this data type.

References: [RFC 3411].

G.2.1.12 PrimaryDsIndicatorType

This data type enumerates the different type of Primary downstream channels. Possible values are:

- primaryDsChannel - when both the CM and CCAP are using DOCSIS 3.1 or DOCSIS 4.0 mode, this value indicates that the channel is the primary channel for the CM receiving this RCC. When DOCSIS 3.0 mode is in use, this value indicates that the channel is primary-capable; multiple such channels are allowed in this mode.
- backupPrimaryDs - when both the CM and CCAP are using DOCSIS 3.1 or DOCSIS 4.0 mode, this value indicates that the channel is a backup primary channel for the CM receiving this RCC. The priority-ordered list of backup primary channels sent to the CM is the same order as the backupPrimaryDs channels are configured in RxChCfg. When DOCSIS 3.0 mode is in use, this value indicates that the channel is primary-capable; DOCSIS 3.0 does not support the backup primary channel feature.
- notSpecified - indicates that this channel has not been specified as a primary-capable channel.
- other - indicates a vendor-specific value.

References: [MULPIv4.0] Receive Channel Primary Downstream Channel Indicator section in the Common Radio Frequency Interface Encodings Annex.

G.2.1.13 RcpId

This data type defines a 'Receive Channel Profile Identifier' (RCP-ID). An RCP-ID consists of 5-octet length string where the first 3-bytes (from left to right corresponds to the Organizational Unique ID (OUI) followed by a two-byte vendor-maintained identifier to represent multiple versions or models of RCP-IDs.

References: [MULPIv4.0] RCP-ID section in the Common Radio Frequency Interface Encodings Annex.

G.2.1.14 ScdmaSelectionString

This data type represents the S-CDMA selection string for active codes used with Selectable Active Codes Mode 2.

A 128-bit string indicating which codes are active. The first element in the string corresponds to code 0 (the all-ones code), and the last element in the string corresponds to code 127. A '1' element in the string indicates an active code, and a '0' indicates an unused code. A zero-length string is returned for an unknown or non-applicable value.

References: [MULPIv4.0] Minislot Numbering Parameters in Timing and Synchronization section.

G.2.1.15 SchedulingType

The scheduling service provided by a CMTS for an upstream Service Flow. This parameter needs to be reported as 'undefined' for downstream QoS Parameter Sets.

Valid enumerations for the data type are:

- undefined(1)
- bestEffort(2)
- nonRealTimePollingService(3)
- realTimePollingService(4)
- unsolicitedGrantServiceWithAD(5)
- unsolicitedGrantService(6)
- proactiveGrantService(7)

Reference: [MULPIv4.0] Service Flow Scheduling Type section.

G.2.2 RCC Status Objects

This section defines the CM Receive Channel Configuration (RCC) Status objects.

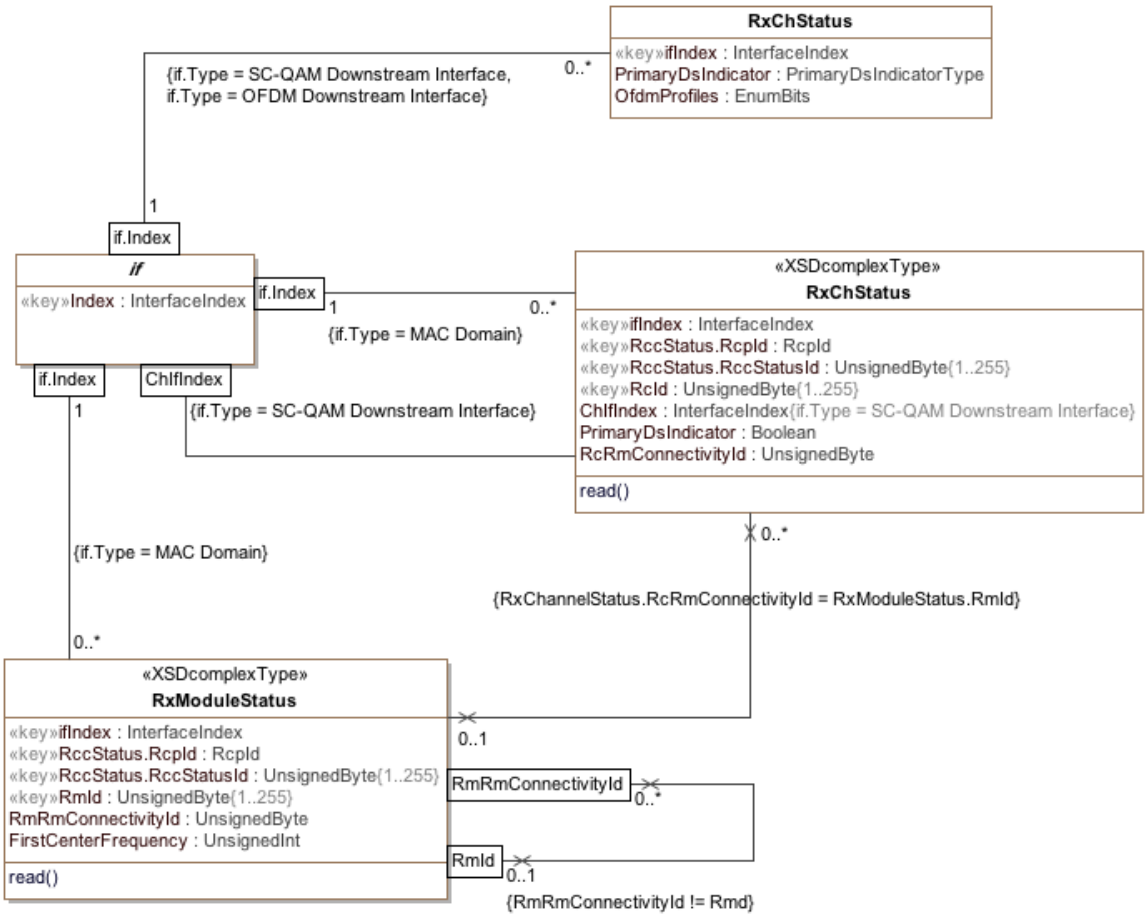


Figure 18 - RCC Status Information Model

G.2.2.1 RxModuleStatus

The Receive Module Status object provides a read-only view of the statically configured and dynamically created Receive Modules within an RCC. When this object is defined on the CM, the value of RccStatusId is always 1.

Table 160 - RxModuleStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	InterfaceIndex of MAC Domain interface		
RcpId	RcpId	Key			
RccStatusId	UnsignedByte	Key	1..255		
RmId	UnsignedByte	Key	1..255		
RmRmConnectivityId	UnsignedByte	R/O			
FirstCenterFrequency	UnsignedInt	R/O		Hz	

G.2.2.1.1 IfIndex

This key represents the interface index of the MAC Domain to which this instance applies.

G.2.2.1.2 RcpId

This key represents the RCP-ID to which this instance applies.

G.2.2.1.3 RccStatusId

This key represents an RCC combination for a particular RcpId either from an RCC configuration object or a CMTS determined RCC and is unique per combination of MAC Domain interface index and RcpId. Note that when this attribute is instantiated at the CM, its value will always be 1.

G.2.2.1.4 RmId

This key represents an identifier of a Receive Module instance within the Receive Channel Profile.

References: [MULPIv4.0] Receive Module Index section in the Common Radio Frequency Interface Encodings Annex.

G.2.2.1.5 RmRmConnectivityId

This attribute represents the Receive Module to which this Receive Module connects. Requirements for module connectivity are detailed in the RmRmConnectivityId of the RccCfg object.

G.2.2.1.6 FirstCenterFrequency

This attribute represents the low frequency channel of the Receive Module, or 0 if not applicable to the Receive Module.

G.2.2.2 Pre-DOCSIS 3.1/4.0 RxChStatus

The Receive Channel Status object reports the status of the statically-configured and dynamically-created Receive Channels within an RCC. When this object is defined on the CM, the value of RccStatusId is always 1.

This object provides the ability for the CM to report its receive channel configuration and is applicable for cases where a DOCSIS 3.1 or DOCSIS 4.0 CM registers with a DOCSIS 3.0 CCAP.

Table 161 - RxChStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	InterfaceIndex of MAC Domain interface		
RcpId	RcpId	Key			
RccStatusId	UnsignedByte	Key	1..255		
RcId	UnsignedByte	Key	1..255		
ChIfIndex	InterfaceIndex	R/O	InterfaceIndex of Downstream Channel assigned to the Receive Channel		
PrimaryDsIndicator	Boolean	R/O			
RcRmConnectivityId	UnsignedByte	R/O			

G.2.2.2.1 IfIndex

This key represents the interface index of the MAC Domain to which this instance applies.

G.2.2.2.2 RcpId

This key represents the RCP-ID to which this instance applies.

G.2.2.2.3 RccStatusId

This key represents an RCC combination for a particular RcpId either from an RCC configuration object or a CMTS determined RCC. It is unique per combination of MAC Domain interface index and RcpId. Note that when this attribute is instantiated at the CM, its value will always be 1.

G.2.2.2.4 RcId

This key represents an identifier for the parameters of the Receive Channel instance within the Receive Channel Profile.

G.2.2.2.5 ChIfIndex

This attribute contains the interface index of the Downstream Channel that this Receive Channel Instance defines.

G.2.2.2.6 PrimaryDsIndicator

If set to 'true', this attribute indicates the Receive Channel is to be the primary-capable downstream channel for the CM receiving this RCC. Otherwise, the downstream channel is to be a non-primary-capable channel.

G.2.2.2.7 RcRmConnectivityId

This attribute identifies the Receive Module to which this Receive Channel connects. A value of zero indicates that the Receive Channel Connectivity TLV is omitted from the RCC.

G.2.2.3 RxChStatus

The Receive Channel Status object reports the status of the statically-configured and dynamically-created Receive Channels within an RCC.

This object provides the ability for the CM to report its receive channel configuration and is applicable for cases where a DOCSIS 4.0 CM registers with a DOCSIS 4.0 CCAP.

Table 162 - RxChStatus Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	SC-QAM or OFDM Index		
PrimaryDsIndicator	PrimaryDsIndicatorType	R/O			
OfdmProfiles	EnumBits	R/O			

G.2.2.3.1 IfIndex

This key represents the SC-QAM or OFDM interface index to which this instance applies.

G.2.2.3.2 PrimaryDsIndicator

This key attribute encodes the type of downstream channel.

G.2.2.3.3 OfdmProfiles

This attribute identifies the downstream channel profiles provisioned on the CM. An example of EnumBits follows.

Example 1: A Cable Modem configured with OFDM Profiles 3 and 8 would return a query response as follows.
EnumBits: 0001000010000000 or 0x1080.

Example 2: A Cable Modem configured with an SC_QAM Channel would return a query response as follows.
EnumBits: 000000000000000 or 0x0.

G.2.3 DOCSIS QoS Objects

This section defines the reporting of the DOCSIS CM QoS configuration. The model is updated in this specification to include objects to configure the Active Queue Management (AQM) feature introduced in [MULPIv4.0].

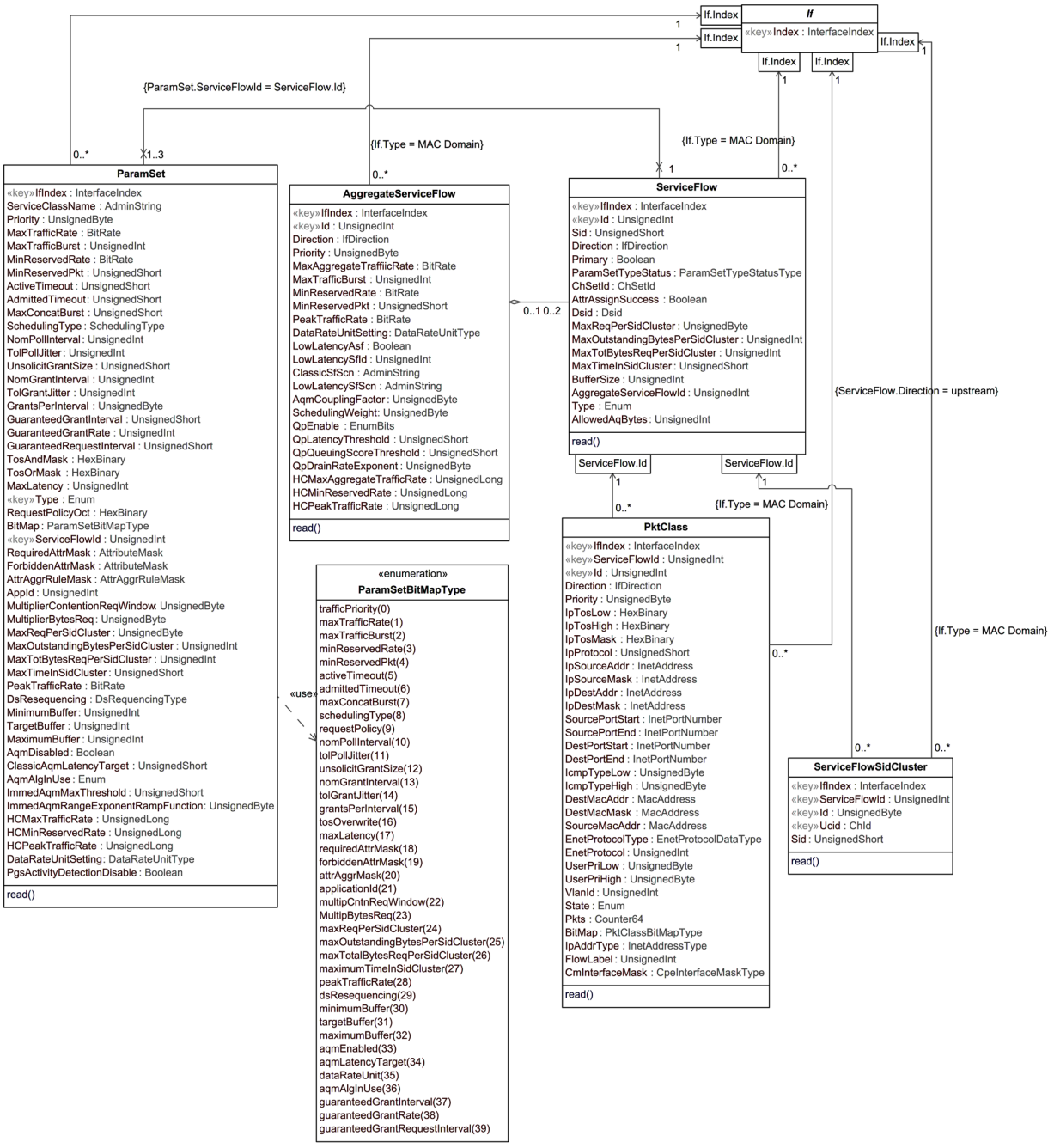


Figure 19 - QoS Configuration Status Information Model

G.2.3.1 PktClass

This object describes the packet classification configured on the CM or CMTS. The model is that a packet either received as input from an interface or transmitted for output on an interface may be compared against an ordered list of rules pertaining to the packet contents. Each rule is an instance of this object. A matching rule provides a Service Flow ID to which the packet is classified. All rules need to match for a packet to match a classifier. The attributes in this row correspond to a set of Classifier Encoding parameters in a DOCSIS MAC management message. The BitMap attribute indicates which particular parameters were present in the classifier as signaled in the DOCSIS message. If the referenced parameter was not present in the signaled Classifier, the corresponding attribute in this instance reports a value as specified by that attribute description.

References: [MULPIv4.0] Service Flows and Classifiers section.

Table 163 - PktClass Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
ServiceFlowId	UnsignedInt	Key	1..4294967295		
Id	UnsignedInt	Key	1..65535		
Direction	IfDirection	R/O			
Priority	UnsignedByte	R/O			
IpTosLow	HexBinary	R/O	SIZE(1)		
IpTosHigh	HexBinary	R/O	SIZE(1)		
IpTosMask	HexBinary	R/O	SIZE(1)		
IpProtocol	UnsignedShort	R/O	0..258		
IpSourceAddr	InetAddress	R/O			
IpSourceMask	InetAddress	R/O			
IpDestAddr	InetAddress	R/O			
IpDestMask	InetAddress	R/O			
SourcePortStart	InetPortNumber	R/O			
SourcePortEnd	InetPortNumber	R/O			
DestPortStart	InetPortNumber	R/O			
DestPortEnd	InetPortNumber	R/O			
IcmpTypeLow	UnsignedByte	R/O			
IcmpTypeHigh	UnsignedByte	R/O			
DestMacAddr	MacAddress	R/O			
DestMacMask	MacAddress	R/O			
SourceMacAddr	MacAddress	R/O			
EnetProtocolType	Enum	R/O			
EnetProtocol	UnsignedInt	R/O	0..65535		
UserPriLow	UnsignedByte	R/O	0..7		
UserPriHigh	UnsignedByte	R/O	0..7		
VlanId	UnsignedInt	R/O	0 | 1..4094		
State	Enum	R/O	active(1) inactive(2)		
Pkts	Counter64	R/O		packets	

Attribute Name	Type	Access	Type Constraints	Units	Default
BitMap	EnumBits	R/O	rulePriority(0), activationState(1), ipTos(2), ipProtocol(3), ipSourceAddr(4), ipSourceMask(5), ipDestAddr(6), ipDestMask(7), sourcePortStart(8), sourcePortEnd(9), destPortStart(10), destPortEnd(11), destMac(12), sourceMac(13), ethertype(14), userPri(15), vlanId(16), flowLabel(17), cmInterfaceMask(18), icmpTypeLow(19), icmpTypeHigh(20)		
IpAddrType	InetAddressType	R/O			
FlowLabel	UnsignedInt	R/O	0..1048575		
CmInterfaceMask	CpeInterfaceMaskType	R/O			

G.2.3.1.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.3.1.2 ServiceFlowId

This key represents an identifier assigned to a Service Flow by CMTS within a MAC Domain. The value 0 is used only for the purpose of reporting instances pertaining UDCs and not used for association of QoS classifiers to service flows.

G.2.3.1.3 Id

This key indicates the assigned identifier to the packet classifier instance by the CMTS, which is unique per Service Flow. For UDCs this corresponds to the Service Flow Reference of the classifier.

References: [MULPIv4.0] Classifier Identifier section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.4 Direction

This attribute indicates the direction to which the classifier is applied.

G.2.3.1.5 Priority

This attribute specifies the order of evaluation of the classifiers. The higher the value, the higher the priority. The value of 0 is used as default in provisioned Service Flows Classifiers. The default value of 64 is used for dynamic Service Flow Classifiers. If the referenced parameter is not present in a classifier, this attribute reports the default value as defined above.

References: [MULPIv4.0] Rule Priority section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.6 IpTosLow

This attribute indicates the low value of a range of TOS byte values. If the referenced parameter is not present in a classifier, this attribute reports the value of 0. The IP TOS octet as originally defined in [RFC 791] has been superseded by the 6-bit Differentiated Services Field (DSField, [RFC 3260]) and the 2-bit Explicit Congestion Notification Field (ECN field, [RFC 3168]). This object is defined as an 8-bit octet as defined by the DOCSIS Specification for packet classification.

References: [MULPIv4.0] IPv4 Type of Service Range and Mask and IPv6 Traffic Class Range and Mask sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.7 IpTosHigh

This attribute indicates the 8-bit high value of a range of TOS byte values. If the referenced parameter is not present in a classifier, this attribute reports the value of 0. The IP TOS octet as originally defined in [RFC 791] has been superseded by the 6-bit Differentiated Services Field (DSField, [RFC 3260]) and the 2-bit Explicit Congestion Notification Field (ECN field, [RFC 3168]). This object is defined as an 8-bit octet as defined by the DOCSIS Specification for packet classification.

References: [MULPIv4.0] IPv4 Type of Service Range and Mask and IPv6 Traffic Class Range and Mask sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.8 IpTosMask

This attribute indicates the mask value is bitwise ANDed with TOS byte in an IP packet, and this value is used for range checking of TosLow and TosHigh. If the referenced parameter is not present in a classifier, this attribute reports the value of 0. The IP TOS octet as originally defined in [RFC 791] has been superseded by the 6-bit Differentiated Services Field (DSField, [RFC 3260]) and the 2-bit Explicit Congestion Notification Field (ECN field, [RFC 3168]). This object is defined as an 8-bit octet per the DOCSIS Specification for packet classification.

References: [MULPIv4.0] IPv4 Type of Service Range and Mask and IPv6 Traffic Class Range and Mask sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.9 IpProtocol

This attribute indicates the value of the IP Protocol field required for IP packets to match this rule. The value 256 matches traffic with any IP Protocol value. The value 257 by convention matches both TCP and UDP. If the referenced parameter is not present in a classifier, this attribute reports the value of 258.

References: [MULPIv4.0] IP Protocol and IPv6 Next Header Type sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.10 IpSourceAddr

This attribute specifies the value of the IP Source Address required for packets to match this rule. An IP packet matches the rule when the packet IP Source Address bitwise ANDed with the IpSourceMask value equals the IpSourceAddr value. The address type of this object is specified by IpAddrType. If the referenced parameter is not present in a classifier, this object reports the value of '00000000'H.

References: [MULPIv4.0] IPv4 Source Address and IPv6 Source Address sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.11 IpSourceMask

This attribute specifies which bits of a packet's IP Source Address are compared to match this rule. An IP packet matches the rule when the packet source address bitwise ANDed with the IpSourceMask value equals the IpSourceAddr value. The address type of this attribute is specified by IpAddrType. If the referenced parameter is not present in a classifier, this attribute reports the value of 'FFFFFFFF'H.

References: [MULPIv4.0] IPv4 Source Mask and IPv6 Source Prefix Length (bits) sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.12 IpDestAddr

This attribute specifies the value of the IP Destination Address required for packets to match this rule. An IP packet matches the rule when the packet IP Destination Address bitwise ANDed with the IpDestMask value equals the IpDestAddr value. The address type of this attribute is specified by IpAddrType. If the referenced parameter is not present in a classifier, this attribute reports the value of '00000000'H.

References: [MULPIv4.0] IPv4 Destination Address and IPv6 Destination Address sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.13 IpDestMask

This attribute specifies which bits of a packet's IP Destination Address are compared to match this rule. An IP packet matches the rule when the packet destination address bitwise ANDed with the IpDestMask value equals the IpDestAddr value. The address type of this attribute is specified by IpAddrType. If the referenced parameter is not present in a classifier, this attribute reports the value of 'FFFFFFFF'H.

References: [MULPIv4.0] IPv4 Destination Mask and IPv6 Destination Prefix Length (bits) sections in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.14 SourcePortStart

This attribute specifies the low-end inclusive range of TCP/UDP source port numbers to which a packet is compared. This attribute is irrelevant for non-TCP/UDP IP packets. If the referenced parameter is not present in a classifier, this attribute reports the value of 0.

References: [MULPIv4.0] TCP/UDP Source Port Start section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.15 SourcePortEnd

This attribute specifies the high-end inclusive range of TCP/UDP source port numbers to which a packet is compared. This attribute is irrelevant for non-TCP/UDP IP packets. If the referenced parameter is not present in a classifier, this attribute reports the value of 65535.

References: [MULPIv4.0] TCP/UDP Source Port End section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.16 DestPortStart

This attribute specifies the low-end inclusive range of TCP/UDP destination port numbers to which a packet is compared. If the referenced parameter is not present in a classifier, this attribute reports the value of 0.

References: [MULPIv4.0] TCP/UDP Destination Port Start section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.17 DestPortEnd

This attribute specifies the high-end inclusive range of TCP/UDP destination port numbers to which a packet is compared. If the referenced parameter is not present in a classifier, this attribute reports the value of 65535.

References: [MULPIv4.0] TCP/UDP Destination Port End section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.18 IcmpTypeLow

This attribute specifies the low-end inclusive range of the ICMP type numbers to which a packet is compared. If the referenced parameter is not present in a classifier, this attribute reports the value of 0.

References: [MULPIv4.0] TypeLow encodings section of the Common Radio Frequency Interface Annex.

G.2.3.1.19 IcmpTypeHigh

This attribute specifies the high-end inclusive range of the ICMP type numbers to which a packet is compared. If the referenced parameter is not present in a classifier, this attribute reports the value of 255.

References: [MULPIv4.0] TypeHigh encodings section of the Common Radio Frequency Interface Annex.

G.2.3.1.20 DestMacAddr

An Ethernet packet matches an entry when its destination MAC address bitwise ANDed with DestMacMask equals the value of DestMacAddr. If the referenced parameter is not present in a classifier, this attribute reports the value of '000000000000'H.

References: [MULPIv4.0] Destination MAC Address section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.21 DestMacMask

An Ethernet packet matches an entry when its destination MAC address bitwise ANDed with DestMacMask equals the value of DestMacAddr. If the referenced parameter is not present in a classifier, this attribute reports the value of '000000000000'H.

References: [MULPIv4.0] Destination MAC Address section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.22 SourceMacAddr

An Ethernet packet matches this entry when its source MAC address equals the value of this attribute. If the referenced parameter is not present in a classifier, this attribute reports the value of 'FFFFFFFFFFFF'.

References: [MULPIv4.0] Source MAC Address section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.23 EnetProtocolType

This attribute indicates the format of the layer 3 protocol ID in the Ethernet packet. A value of 'none' means that the rule does not use the layer 3 protocol type as a matching criteria. A value of 'ethertype' means that the rule applies only to frames that contain an EtherType value. Ethertype values are contained in packets using the Dec-Intel-Xerox (DIX) encapsulation or the RFC1042 Sub-Network Access Protocol (SNAP) encapsulation formats. A value of 'dsap' means that the rule applies only to frames using the IEEE802.3 encapsulation format with a Destination Service Access Point (DSAP) other than 0xAA (which is reserved for SNAP). A value of 'mac' means that the rule applies only to MAC management messages for MAC management messages. A value of 'all' means that the rule matches all Ethernet packets. If the Ethernet frame contains an 802.1P/Q Tag header (i.e., EtherType 0x8100), this attribute applies to the embedded EtherType field within the 802.1P/Q header. If the referenced parameter is not present in a classifier, this attribute reports the value of 0.

References: [MULPIv4.0] Ethertype/DSAP/MacType section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.24 EnetProtocol

If EnetProtocolType is 'none', this attribute is ignored when considering whether a packet matches the current rule. If EnetProtocolType is 'ethertype', this attribute gives the 16-bit value of the EtherType that the packet needs to match in order to match the rule. If EnetProtocolType is 'dsap', the lower 8 bits of this attribute's value needs to match the DSAP byte of the packet in order to match the rule. If EnetProtocolType is 'mac', the lower 8 bits of this attribute's value represent a lower bound (inclusive) of MAC management message type codes matched, and the upper 8 bits represent the upper bound (inclusive) of matched MAC message type codes. Certain message type codes are excluded from matching, as specified in the reference. If the Ethernet frame contains an 802.1P/Q Tag header (i.e., EtherType 0x8100), this attribute applies to the embedded EtherType field within the 802.1P/Q header. If the referenced parameter is not present in the classifier, the value of this attribute is reported as 0.

References: [MULPIv4.0] Ethertype/DSAP/MacType section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.25 UserPriLow

This attribute applies only to Ethernet frames using the 802.1P/Q tag header (indicated with EtherType 0x8100). Such frames include a 16-bit Tag that contains a 3-bit Priority field and a 12-bit VLAN number. Tagged Ethernet packets need to have a 3-bit Priority field within the range of PriLow to PriHigh in order to match this rule. If the referenced parameter is not present in the classifier, the value of this attribute is reported as 0.

References: [MULPIv4.0] IEEE 802.1P User_Priority section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.26 UserPriHigh

This attribute applies only to Ethernet frames using the 802.1P/Qtag header (indicated with EtherType 0x8100). Such frames include a 16-bit Tag that contains a 3-bit Priority field and a 12-bit VLAN number. Tagged Ethernet packets need to have a 3-bit Priority field within the range of PriLow to PriHigh in order to match this rule. If the referenced parameter is not present in the classifier, the value of this attribute is reported as 7.

References: [MULPIv4.0] IEEE 802.1P User_Priority section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.27 VlanId

This attribute applies only to Ethernet frames using the 802.1P/Q tag header. Tagged packets need to have a VLAN Identifier that matches the value in order to match the rule. If the referenced parameter is not present in the classifier, the value of this attribute is reported as 0.

References: [MULPIv4.0] IEEE 802.1Q VLAN_ID section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.28 State

This attribute indicates whether or not the classifier is enabled to classify packets to a Service Flow. If the referenced parameter is not present in the classifier, the value of this attribute is reported as 'true'.

References: [MULPIv4.0] Classifier Activation State section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.29 Pkts

This attribute counts the number of packets that have been classified using this entry. This includes all packets delivered to a Service Flow maximum rate policing function, whether or not that function drops the packets. This counter's last discontinuity is the ifCounterDiscontinuityTime for the same ifIndex that indexes this attribute.

G.2.3.1.30 BitMap

This attribute indicates which parameter encodings were actually present in the DOCSIS packet classifier encoding signaled in the DOCSIS message that created or modified the classifier. Note that Dynamic Service Change messages have replace semantics, so that all non-default parameters need to be present whether the classifier is being created or changed. A bit of this attribute is set to 1 if the parameter indicated by the comment was present in the classifier encoding, and to 0 otherwise. Note that BITS are encoded most significant bit first, so that if, for example, bits 6 and 7 are set, this attribute is encoded as the octet string '030000'H.

G.2.3.1.31 IpAddrType

This attribute indicates the type of the Internet address for IpSourceAddr, IpSourceMask, IpDestAddr, and IpDestMask. If the referenced parameter is not present in a classifier, this object reports the value of 'ipv4'.

G.2.3.1.32 FlowLabel

This attribute represents the Flow Label field in the IPv6 header to be matched by the classifier. The value zero indicates that the Flow Label is not specified as part of the classifier and is not matched against the packets.

References: [MULPIv4.0] IPv6 Flow Label section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.1.33 CmInterfaceMask

This attribute represents a bit-mask of the CM in-bound interfaces to which this classifier applies. This attribute only applies to QoS upstream Classifiers and upstream Drop Classifiers. For QoS downstream classifiers this object reports the zero-length string.

References: [MULPIv4.0] CM Interface Mask (CMIM) Encoding section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2 ParamSet

This object describes the set of QoS parameters defined in a managed device. Each row corresponds to a DOCSIS QoS Parameter Set as signaled via DOCSIS MAC management messages. Each attribute of an instance of this object corresponds to one or part of one Service Flow Encoding. The BitMap attribute indicates which particular parameters were signaled in the original registration or dynamic service request message that created the QoS Parameter Set. In many cases, even if a QoS Parameter Set parameter was not signaled, the DOCSIS specification calls for a default value to be used. That default value is reported as the value of the corresponding attribute in this object instance. Many attributes are not applicable, depending on the Service Flow direction, upstream scheduling type or Service Flow bonding configuration. The attribute value reported in this case is specified by those attributes descriptions.

For CM devices supporting the ParamSet object, the CM MUST report only the active service flow parameter set values. The CM MAY report the admitted and provisioned service flow parameter sets.

References: [MULPIv4.0] Service Flow Encodings section in the Common Radio Frequency Interface Encodings Annex.

Table 164 - ParamSet Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default (See Attribute Description)
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
ServiceClassName	AdminString	R/O	SIZE (0..15)		
Priority	UnsignedByte	R/O	0..7		
MaxTrafficRate	BitRate	R/O			
MaxTrafficBurst	UnsignedInt	R/O		bytes	
MinReservedRate	BitRate	R/O			
MinReservedPkt	UnsignedShort	R/O		bytes	
ActiveTimeout	UnsignedShort	R/O		seconds	
AdmittedTimeout	UnsignedShort	R/O		seconds	
MaxConcatBurst	UnsignedShort	R/O		bytes	
SchedulingType	SchedulingType	R/O			
NomPollInterval	UnsignedInt	R/O		microseconds	
TolPollJitter	UnsignedInt	R/O		microseconds	
UnsolicitGrantSize	UnsignedShort	R/O		bytes	
NomGrantInterval	UnsignedInt	R/O		microseconds	
TolGrantJitter	UnsignedInt	R/O		microseconds	
GrantsPerInterval	UnsignedByte	R/O	0..127	dataGrants	

Attribute Name	Type	Access	Type Constraints	Units	Default (See Attribute Description)
GuaranteedGrantInterval	UnsignedShort	R/O		microseconds	
GuaranteedGrantRate	BitRate	R/O			
GuaranteedGrantRequestInterval	UnsignedShort	R/O		microseconds	
TosAndMask	HexBinary	R/O	SIZE (1)		
TosOrMask	HexBinary	R/O	SIZE (1)		
MaxLatency	UnsignedInt	R/O		microseconds	
Type	Enum	Key	active (1) admitted (2) provisioned (3)		
RequestPolicyOct	HexBinary	R/O	SIZE (4)		
BitMap	EnumBits	R/O	trafficPriority(0) maxTrafficRate(1) maxTrafficBurst(2) minReservedRate(3) minReservedPkt(4) activeTimeout(5) admittedTimeout(6) maxConcatBurst(7) schedulingType(8) requestPolicy(9) nomPollInterval(10) tolPollJitter(11) unsolicitGrantSize(12) nomGrantInterval(13) tolGrantJitter(14) grantsPerInterval(15) tosOverwrite(16) maxLatency(17) requiredAttrMask(18) forbiddenAttrMask(19) attrAggrMask(20) applicationId(21) multipCntnReqWindow(22) multipBytesReq(23) maxReqPerSidCluster(24) maxOutstandingBytesPerSidCluster(25) maxTotalBytesReqPerSidCluster(26) maximumTimeInSidCluster(27) peakTrafficRate(28) dsResequencing(29) minimumBuffer(30) targetBuffer(31) maximumBuffer(32) aqmEnabled(33) aqmLatencyTarget(34) dataRateUnit(35) aqmAlgInUse(36) guaranteedGrantInterval(37) guaranteedGrantRate(38) guaranteedGrantRequestInterval(39) immedAqmMaxThrshld(40) immedAqmRngExpRampFunc(41) pgsActivityDetectionDisable(42)		
ServiceFlowId	UnsignedInt	Key	1.. 4294967295		
RequiredAttrMask	AttributeMask	R/O			
ForbiddenAttrMask	AttributeMask	R/O			
AttrAggrRuleMask	AttrAggrRuleMask	R/O	SIZE (0 | 4)		
AppId	UnsignedInt	R/O			
MultiplierContentionReqWindow	UnsignedByte	R/O	4..12	eighths	

Attribute Name	Type	Access	Type Constraints	Units	Default (See Attribute Description)
MultiplierBytesReq	UnsignedByte	R/O	1 | 2 | 4 | 8 | 16	requests	
MaxReqPerSidCluster	UnsignedByte	R/O		bytes	
MaxOutstandingBytesPerSidCluster	UnsignedInt	R/O		bytes	
MaxTotBytesReqPerSidCluster	UnsignedInt	R/O		bytes	
MaxTimeInSidCluster	UnsignedShort	R/O		milliseconds	
PeakTrafficRate	BitRate	R/O			
DsResequencing	Enum	R/O	resequencingDsidIfBonded(0) noResequencingDsid(1) notApplicable(2)		
MinimumBuffer	UnsignedInt	R/O	0..4294967295	bytes	
TargetBuffer	UnsignedInt	R/O	0..4294967295	bytes	
MaximumBuffer	UnsignedInt	R/O	0..4294967295	bytes	
AqmDisabled	Boolean	R/O			
ClassicAqmLatencyTarget	UnsignedShort	R/O	0..256	msec	
AqmAlgInUse	Enum	R/O	unknown(1) other(2) docsisPIE(3) immediateAqm(4)		
ImmedAqmMaxThreshold	UnsignedShort	R/O			
ImmedAqmRangeExponentRampFunction	UnsignedByte	R/O			
DataRateUnitSetting	DataRateUnitType	R/O		N/A	'bps'

G.2.3.2.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.3.2.2 ServiceClassName

This attribute represents the Service Class Name from which the parameter set values were derived. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns the zero-length string.

References: [MULPIv4.0] Service Class Name section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.3 Priority

This attribute represents the relative priority of a Service Flow. Higher numbers indicate higher priority. This priority should only be used to differentiate Service Flow from identical parameter sets. This attribute returns 0 if the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set or if the parameter is not applicable.

References: [MULPIv4.0] Traffic Priority section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.4 MaxTrafficRate

This attribute represents the 4-byte value of the maximum sustained traffic rate allowed for this Service Flow. It represents all MAC frame data PDUs from the bytes following the MAC header HCS to the end of the CRC. The number of bytes forwarded is limited during any time interval. The value 0 means no maximum traffic rate is enforced. The value of the DataRateUnitSetting attribute defines the units of MaxTrafficRate. This attribute applies to both upstream and downstream Service Flows. This attribute returns 0 if the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, or if the parameter is not applicable.

References: [MULPIv4.0] Maximum Sustained Traffic Rate section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.5 MaxTrafficBurst

This attribute specifies the token bucket size in bytes for this parameter set. The value is calculated from the byte following the MAC header HCS to the end of the CRC. This object is applied in conjunction with MaxTrafficRate to calculate maximum sustained traffic rate. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns 3044 for scheduling types 'bestEffort', 'nonRealTimePollingService' and 'realTimePollingService'. If this parameter is not applicable, it is reported as 0.

References: [MULPIv4.0] Maximum Traffic Burst section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.6 MinReservedRate

This attribute represents the 4-byte value of the guaranteed minimum rate for this Service Flow. The value is calculated from the byte following the MAC header HCS to the end of the CRC. The value of 0 indicates that no bandwidth is reserved. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns 0. If the parameter is not applicable, it is reported as 0. The value of the DataRateUnitSetting attribute defines the units of MinReservedRate.

References: [MULPIv4.0] Minimum Reserved Traffic Rate section of the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.7 MinReservedPkt

This attribute specifies an assumed minimum packet size in bytes for which the MinReservedRate will be provided. The value is calculated from the byte following the MAC header HCS to the end of the CRC. If the referenced parameter is omitted from a DOCSIS QoS parameter set, the used and reported value is CMTS implementation and the CM reports a value of 0. If the referenced parameter is not applicable to the direction or scheduling type of the Service Flow, both CMTS and CM report the value 0.

References: [MULPIv4.0] Assumed Minimum Reserved Rate Packet Size, in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.8 ActiveTimeout

This attribute specifies the maximum duration in seconds that resources remain unused on an active service flow before the CMTS signals that both the active and admitted parameter sets are null. The value 0 signifies an infinite amount of time. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns 0.

References: [MULPIv4.0] Timeout for Active QoS Parameters section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.9 AdmittedTimeout

This attribute specifies the maximum duration in seconds that resources remain in admitted state before resources need to be released. The value of 0 signifies an infinite amount of time. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns 200.

References: [MULPIv4.0] Timeout for Admitted QoS Parameters section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.10 MaxConcatBurst

This attribute specifies the maximum concatenated burst in bytes that an upstream Service Flow is allowed. The value is calculated from the FC byte of the Concatenation MAC Header to the last CRC byte of the last concatenated MAC frame, inclusive. The value of 0 specifies no maximum burst. If the referenced parameter is not present in the

corresponding DOCSIS QoS Parameter Set, this attribute returns the value of 1522 for scheduling types 'bestEffort', 'nonRealTimePollingService', and 'realTimePollingService'. If the parameter is not applicable, it is reported as 0.

References: [MULPIv4.0] Maximum Concatenated Burst section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.11 SchedulingType

This attribute specifies the upstream scheduling service used for upstream Service Flow. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set of an upstream Service Flow, this attribute returns the value of 'bestEffort'. For QoS parameter sets of downstream Service Flows, this attribute's value is reported as 'undefined'.

References: [MULPIv4.0] Service Flow Scheduling Type section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.12 NomPollInterval

This attribute specifies the nominal interval in microseconds between successive unicast request opportunities on an upstream Service Flow. This attribute applies only to upstream Service Flows with SchedulingType of value 'nonRealTimePollingService', 'realTimePollingService', and 'unsolictedGrantServiceWithAD'. The parameter is mandatory for 'realTimePollingService'. If the parameter is omitted with 'nonRealTimePollingService', the CMTS uses an implementation-dependent value. If the parameter is omitted with 'unsolictedGrantServiceWithAD(5)' the CMTS uses the value of the Nominal Grant Interval parameter. In all cases, the CMTS reports the value it is using when the parameter is applicable. The CM reports the signaled parameter value if it was signaled. Otherwise, it returns 0. If the referenced parameter is not applicable to the direction or scheduling type of the corresponding DOCSIS QoS Parameter Set, both CMTS and CM report this attribute's value as 0.

References: [MULPIv4.0] Polling Interval section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.13 TolPollJitter

This attribute specifies the maximum amount of time in microseconds that the unicast request interval may be delayed from the nominal periodic schedule on an upstream Service Flow. This parameter is applicable only to upstream Service Flows with a SchedulingType of 'realTimePollingService' or 'unsolictedGrantServiceWithAD'. If the referenced parameter is applicable but not present in the corresponding DOCSIS QoS Parameter Set, the CMTS uses an implementation-dependent value and reports the value it is using. The CM reports a value of 0 in this case. If the parameter is not applicable to the direction or upstream scheduling type of the Service Flow, both CMTS and CM report this attribute's value as 0.

References: [MULPIv4.0] Tolerated Poll Jitter section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.14 UnsolicitGrantSize

This attribute specifies the unsolicited grant size in bytes. The grant size includes the entire MAC frame data PDU from the Frame Control byte to the end of the MAC frame. The referenced parameter is applicable only for upstream flows with a SchedulingType of 'unsolicitedGrantServiceWithAD' or 'unsolicitedGrantService', and it is mandatory when applicable. Both CMTS and CM report the signaled value of the parameter in this case. If the referenced parameter is not applicable to the direction or scheduling type of the corresponding DOCSIS QoS Parameter Set, both CMTS and CM report this attribute's value as 0.

References: [MULPIv4.0] Unsolicited Grant Size section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.15 NomGrantInterval

This attribute specifies the nominal interval in microseconds between successive data grant opportunities on an upstream Service Flow. The referenced parameter is applicable only for upstream flows with a SchedulingType of 'unsolicitedGrantServiceWithAD' or 'unsolicitedGrantService(6)', and it is mandatory when applicable. Both CMTS and CM report the signaled value of the parameter in this case. If the referenced parameter is not applicable to the

direction or scheduling type of the corresponding DOCSIS QoS Parameter Set, both CMTS and CM report this attribute's value as 0.

References: [MULPIv4.0] Nominal Grant Interval section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.16 TolGrantJitter

This attribute specifies the maximum amount of time in microseconds that the transmission opportunities may be delayed from the nominal periodic schedule. The referenced parameter is applicable only for upstream flows with a SchedulingType of 'unsolicitedGrantServiceWithAD' or 'unsolicitedGrantService(6)', and it is mandatory when applicable. Both CMTS and CM report the signaled value of the parameter in this case. If the referenced parameter is not applicable to the direction or scheduling type of the corresponding DOCSIS QoS Parameter Set, both CMTS and CM report this attribute's value as 0.

References: [MULPIv4.0] Tolerated Grant Jitter section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.17 GrantsPerInterval

This attribute specifies the number of data grants per Nominal Grant Interval (NomGrantInterval). The referenced parameter is applicable only for upstream flows with a SchedulingType of 'unsolicitedGrantServiceWithAD' or 'unsolicitedGrantService', and it is mandatory when applicable. Both CMTS and CM report the signaled value of the parameter in this case. If the referenced parameter is not applicable to the direction or scheduling type of the corresponding DOCSIS QoS Parameter Set, both CMTS and CM report this attribute's value as 0.

References: [MULPIv4.0] Grants per Interval section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.18 GuaranteedGrantInterval

This attribute specifies the maximum interval between successive data transmission opportunities for a PGS Service Flow. This attribute is a 16-bit representation of the grant interval in units of microseconds.

References: [MULPIv4.0] Guaranteed Grant Interval section in the Common TLV Encodings Annex.

G.2.3.2.19 GuaranteedGrantRate

This attribute specifies the minimum granting rate for an upstream PGS service flow. The value of the DataRateUnitSetting attribute defines the units of GuaranteedGrantRate.

References: [MULPIv4.0] Guaranteed Grant Rate section in the Common TLV Encodings Annex.

G.2.3.2.20 GuaranteedRequestInterval

The value of this parameter specifies the maximum interval between successive request opportunities (including unicast request opportunities and piggyback request opportunities) for an upstream PGS service flow. This attribute is a 16-bit representation of the request interval in units of microseconds. The value zero represents polling disabled.

References: [MULPIv4.0] Guaranteed Request Interval section in the Common TLV Encodings Annex.

G.2.3.2.21 TosAndMask

This attribute specifies the AND mask for the IP TOS byte for overwriting an IPv4 packet's TOS value or IPv6 packet's Traffic Class value. The IP packet TOS byte is bitwise ANDed with TosAndMask, then the result is bitwise ORed with TosORMask and the result is written to the IP packet TOS byte. A value of 'FF'H for TosAndMask and a value of '00'H for TosOrMask means that the IP Packet TOS byte is not overwritten. This combination is reported if the referenced parameter is not present in a QoS Parameter Set. The IP TOS octet as originally defined in [RFC 791] has been superseded by the 6-bit Differentiated Services Field (DSField, [RFC 3260]) and the 2-bit Explicit Congestion Notification Field (ECN field, [RFC 3168]). The IPv6 Traffic Class octet [RFC 2460] is consistent with that new definition. Network operators should avoid specifying values of TosAndMask and TosORMask that would result in the modification of the ECN bits. In particular, operators should not use values of TosAndMask that have

either of the least-significant two bits set to 0. Similarly, operators should not use values of TosORMask that have either of the least-significant two bits set to 1. Even though this attribute is only enforced by the CMTS, the CM reports the value as signaled in the referenced parameter.

References: [MULPIv4.0] IP Type Of Service (DSCP) Overwrite section in the Common Radio Frequency Interface Encodings Annex; [RFC 3168]; [RFC 3260]; [RFC 2460]; [RFC 791].

G.2.3.2.22 TosOrMask

This attribute specifies the OR mask for the IPv4 TOS value or IPv6 Traffic Class value. See the description of TosAndMask for further details. The IP TOS octet, as originally defined in [RFC 791] has been superseded by the 6-bit Differentiated Services Field (DSField, [RFC 3260]) and the 2-bit Explicit Congestion Notification Field (ECN field, [RFC 3168]). The IPv6 Traffic Class octet [RFC 2460] is consistent with that new definition. Network operators should avoid specifying values of TosAndMask and TosORMask that would result in the modification of the ECN bits.

References: [MULPIv4.0] IP Type Of Service (DSCP) Overwrite section in the Common Radio Frequency Interface Encodings Annex; [RFC 3168]; [RFC 3260]; [RFC 2460]; [RFC 791].

G.2.3.2.23 MaxLatency

This attribute specifies the maximum latency between the reception of a packet by the CMTS on its NSI and the forwarding of the packet to the RF interface. A value of 0 signifies no maximum latency is enforced. This attribute only applies to downstream Service Flows. If the referenced parameter is not present in the corresponding downstream DOCSIS QoS Parameter Set, this attribute returns 0. This parameter is not applicable to upstream DOCSIS QoS Parameter Sets, so its value is reported as 0 in that case.

References: [MULPIv4.0] Maximum Downstream Latency section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.24 Type

This key represents the QoS Parameter Set Type of the Service Flow. The following values are defined: 'active' Indicates the Active QoS parameter set, describing the service currently being provided by the DOCSIS MAC domain to the service flow. 'admitted' indicates the Admitted QoS Parameter Set, describing services reserved by the DOCSIS MAC domain for use by the service flow. 'provisioned' indicates the QoS Parameter Set defined in the DOCSIS CM Configuration file for the service flow.

Only the 'active' service flow parameter set is required to be reported. The 'admitted' and 'provisioned' sets are not required to be reported.

References: [MULPIv4.0] Service Flow Scheduling Type section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.25 RequestPolicyOct

This attribute specifies which transmit interval opportunities the CM omits for upstream transmission requests and packet transmissions. This object takes its default value for downstream Service Flows. Unless otherwise indicated, a bit value of 1 means that a CM cannot use that opportunity for upstream transmission. The format of this string enumerated the bits from 0 to 31 from left to right, for example bit 0 corresponds to the left most bit of the fourth octet. (octets numbered from right to left). The bit positions are defined as follows:

'broadcastReqOpp' - all CMs broadcast request opportunities

'priorityReqMulticastReq' - priority request multicast request opportunities

'reqDataForReq' - request/data opportunities for requests

'reqDataForData' - request/data opportunities for data

'piggybackReqWithData' - piggyback requests with data

'concatenateData' - concatenate data

'fragmentData' - fragment data

'suppressPayloadHeaders' - suppress payload headers

'dropPktsExceedUGSize' - A value of 1 means that the service flow needs to drop packets that do not fit in the Unsolicited Grant size. If the referenced parameter is not present in a QoS Parameter Set, the value of this object is reported as '00000000'H.

References: [MULPIv4.0] Request/ Transmission Policy section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.26 BitMap

This attribute indicates the set of QoS Parameter Set parameters actually signaled in the DOCSIS registration or dynamic service request message that created or modified the QoS Parameter Set. Possible QoS Parameter Set parameters are listed below with the corresponding TLV Type. A bit is set to 1 when the associated parameter is present in the original request as follows:

'trafficPriority' Traffic Priority (24/25.7)

'maxTrafficRate' Maximum Sustained Traffic Rate (24/25.8)

'maxTrafficBurst' Maximum Traffic Burst (24/25.9)

'minReservedRate' Minimum Reserved Traffic Rate (24/25.10)

'minReservedPkt' Assumed Minimum Reserved Rate Packet Size (24/25.11)

'activeTimeout' Timeout for Active QoS Parameters (24/25.12)

'admittedTimeout' Timeout for Admitted QoS Parameters (24/25.13)

'maxConcatBurst' Maximum Concatenated Burst (24.14)

'schedulingType' Service Flow Scheduling Type (24.15)

'requestPolicy' Request/Transmission Policy (24.16)

'nomPollInterval' Nominal Polling Interval (24.17)

'tolPollJitter' Tolerated Poll Jitter (24.18)

'unsolicitGrantSize' Unsolicited Grant Size (24.19)

'nomGrantInterval' Nominal Grant Interval (24.20)

'tolGrantJitter' Tolerated Grant Jitter (24.21)

'grantsPerInterval' Grants per Interval (24.22)

'tosOverwrite' IP Type of Service (DSCP) Overwrite (24.23)

'maxLatency' Maximum Downstream Latency (25.14)

'requiredAttrMask' Service Flow Required Attribute Mask (24/25.31)

'forbiddenAttrMask Service Flow Forbidden Attribute Mask (24/25.32)

'attrAggrMask' Service Flow Attribute Aggregation Mask (24/25.33)

'applicationId' Application Identifier (24/25.34)

'multipCntnReqWindow' Multiplier to Contention Request Backoff Window(24.25)

'multipBytesReq' Multiplier to Number of Bytes Requested (24.26)

'maxReqPerSidCluster' Maximum Requests per SID Cluster (47/89.3.1)

'maxOutstandingBytesPerSidCluster' Maximum Outstanding Bytes per SID Cluster (47/89.3.2)

'maxTotalBytesReqPerSidCluster' Maximum Total Bytes Requested per SID Cluster(47/89.3.3)

'maximumTimeInSidCluster' Maximum Time in the SID Cluster (47/89.3.4)

'peakTrafficRate' Peak Traffic Rate (24/25.27)

'dsResequencing' - Downstream Resequencing (25.17)

'minimumBuffer' Minimum Buffer (24/25.35.1)

'targetBuffer' Target Buffer (24/25.35.2)

'maximumBuffer' Maximum Buffer (24/25.35.3)

'aqmDisabled' SF AQM Disabled (24/25.40.1)

'classicAqmLatencyTarget' Classic AQM Latency Target (24/25.40.2)

'dataRateUnit' Data Rate Unit Setting (24/25/70/71.41)

'aqmAlgInUse' AQM Algorithm (24/25.40.3)

'guaranteedGrantInterval' Guaranteed Grant Interval (24.44)

'guaranteedGrantRate' Guaranteed Grant Rate (24.45)

'guaranteedGrantRequestInterval' Guaranteed Request Interval (24.46)

'immedAqmMaxThrshld' Immediate AQM Maximum Threshold (24/25.40.4)

'immedAqmRngExpRampFunc' Immediate AQM Range Exponent of Ramp Function (24/25.40.5)

'pgsActivityDetectionDisable' PGS Activity Detection Disable (24.49)

Note that when Service Class names are expanded, the registration or dynamic response message may contain parameters expanded by the CMTS based on a stored service class. These expanded parameters are not indicated by a 1 bit in this attribute. Note that even though some QoS Parameter Set parameters may not be signaled in a message (so that the parameter's bit in this object is 0), the DOCSIS specification requires that default values be used. These default values are reported as the corresponding attribute.

References: [MULPIv4.0] Service Flow Encodings section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.27 ServiceFlowId

This key represents the Service Flow ID for the service flow.

References: [MULPIv4.0] Service Identifier section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.28 RequiredAttrMask

This attribute specifies the Required Attribute Mask to compare with the Provisioned Required Attributes when selecting the bonding groups for the service flow.

If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns '00000000'H.

References: [MULPIv4.0] Service Flow Required Attribute Mask section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.29 ForbiddenAttrMask

This attribute specifies the Forbidden Attribute Mask to compare with the Provisioned Forbidden Attributes when selecting the bonding groups for the service flow.

References: [MULPIv4.0] Service Flow Forbidden Attribute Mask section in the Common Radio Frequency Interface Encodings Annex.

If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns '00000000'H.

G.2.3.2.30 AttrAggrRuleMask

This attribute specifies the Attribute Aggregation Mask to compare the Service Flow Required and Forbidden Attributes with the CMTS dynamically-created bonding group when selecting the bonding groups for the service flow.

References: [MULPIv4.0] Service Flow Attribute Aggregation Mask section in the Common Radio Frequency Interface Encodings Annex.

If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns '00000000'H.

G.2.3.2.31 AppId

This attribute represents the Application Identifier associated with the service flow for purposes beyond the scope of this specification.

If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns 0.

References: [MULPIv4.0] Application Identifier section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.32 MultiplierContentionReqWindow

This attribute specifies the multiplier to be applied by a CM when performing contention request backoff for data requests. This attribute only applies to upstream Service Flows in 3.0 operation. if the referenced parameter is not present in the upstream DOCSIS QoS Parameter Set, or is not applicable, this attribute returns 8.

References: [MULPIv4.0] Multiplier to Contention Request Backoff Window section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.33 MultiplierBytesReq

This attribute specifies the assumed bandwidth request multiplier. This attribute only applies to upstream Service Flows in 3.0 operation. If the referenced parameter is not present in the upstream DOCSIS QoS Parameter Set, or is not applicable, this attribute returns 4.

References: [MULPIv4.0] Multiplier to Number of Bytes Requested section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.34 MaxReqPerSidCluster

This attribute specifies the maximum number of requests that a CM can make within a given SID Cluster before it needs to switch to a different SID Cluster to make further requests. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0. If the referenced parameter is not present in the DOCSIS QoS Parameter Set, this attribute returns 0.

Note: This attribute has been deprecated and replaced with MaxReqPerSidCluster in the ServiceFlow object.

References: [MULPIv4.0] Maximum Requests per SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.35 MaxOutstandingBytesPerSidCluster

This attribute specifies the maximum number of bytes for which a CM can have requests outstanding on a given SID Cluster. If defined number of bytes are outstanding and further requests are required, the CM needs to switch to a different SID Cluster if one is available. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0. If the referenced parameter is not present in the DOCSIS QoS Parameter Set, this attribute returns 0.

Note: This attribute has been deprecated and replaced with MaxOutstandingBytesPerSidCluster in the ServiceFlow object.

References: [MULPIv4.0] Maximum Outstanding Bytes per SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.36 MaxTotBytesReqPerSidCluster

This attribute specifies the maximum total number of bytes a CM can have requested using a given SID Cluster before it needs to switch to a different SID Cluster to make further requests. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0. If the referenced parameter is not present in the DOCSIS QoS Parameter Set, this attribute returns 0.

Note: This attribute has been deprecated and replaced with MaxTotBytesReqPerSidCluster in the ServiceFlow object.

References: [MULPIv4.0] Maximum Total Bytes Requested per SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.37 MaxTimeInSidCluster

This attribute specifies the maximum time in milliseconds that a CM may use a particular SID Cluster before it needs to switch to a different SID Cluster to make further requests. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0. If the referenced parameter is not present in the DOCSIS QoS Parameter Set, this attribute returns 0.

Note: This attribute has been deprecated and replaced with MaxTimeInSidCluster in the ServiceFlow object.

References: [MULPIv4.0] Maximum Time in the SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.38 PeakTrafficRate

This attribute represents the 4-byte value of the rate parameter 'P' of a token-bucket-based peak rate limiter for packets of a service flow. A value of 0 signifies no Peak Traffic Rate is enforced. If the referenced parameter is not present in the corresponding DOCSIS QoS Parameter Set, this attribute returns 0. The value of the DataRateUnitSetting attribute defines the units of PeakTrafficRate.

References: [MULPIv4.0] Peak Traffic Rate section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.39 DsResequencing

This attribute specifies if a resequencing DSID needs to be allocated to the service flow.

The value 'notApplicable' indicates the value of this attribute is not applicable.

The value 'resequencingDsid' indicates that a resequencing DSID is required if the service flow is assigned to a downstream bonding group

The value 'noResequencingDsid' indicates no resequencing DSID is associated with the service flow.

This attribute only applies to downstream Service Flows in 3.0 operation. If the referenced parameter is not present in the corresponding downstream DOCSIS QoS Parameter Set, this attribute returns 'notApplicable'. This parameter is not applicable to upstream DOCSIS QoS Parameter Sets, so the value 'notApplicable' is reported in that case.

References: [MULPIv4.0] Downstream Resequencing section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.40 MinimumBuffer

This attribute represents the configured minimum buffer size for the service flow.

References: [MULPIv4.0] Buffer Control section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.41 TargetBuffer

This attribute represents the configured target buffer size for the service flow. The value 0 indicates that no target buffer size was configured, and the device will use a vendor-specific value.

References: [MULPIv4.0] Buffer Control section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.42 MaximumBuffer

This attribute represents the configured maximum buffer size for the service flow. The value 4294967295 indicates that no maximum buffer size was configured, and thus there is no limit to the buffer size.

References: [MULPIv4.0] Buffer Control section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.43 AqmDisabled

If this attribute is set to 'true', AQM is disabled on the upstream or downstream service flow specified by ServiceFlowId.

References: [MULPIv4.0] AQM Encodings section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.44 ClassicAqmLatencyTarget

This attribute provides the target latency for this service flow when operating under Classic Active Queue Management (e.g., DOCSIS-PIE). This parameter will be ignored if the AQM Algorithm used by the Service Flow is ImmediateAqm. For downstream service flows, the value 256 indicates an unknown latency target. The units are in milliseconds.

References: [MULPIv4.0] AQM Encodings section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.2.45 AqmAlgInUse

This attribute specifies the AQM algorithm in use for this service flow. If AQM is disabled on the service flow specified by ServiceFlowId, this attribute returns unknown(1).

The value unknown(1) is reported for downstream service flows or when AQM is disabled.

The value other(2) indicates a vendor proprietary algorithm for upstream queue management.

The value docsisPIE(3) indicates the upstream queue management Proportional Integral controller Enhanced (PIE) algorithm.

The value immediateAqm(4) indicates the use of the Immediate AQM algorithm.

References: [MULPIv4.0] Proportional-Integral-Enhanced Active Queue Management Algorithm Annex and Immediate Active Queue Management Annex.

G.2.3.2.46 ImmedAqmMaxThreshold

This attribute specifies the maximum threshold in microseconds of the ramp function used by the Immediate AQM algorithm and the Queue Protection algorithm. This attribute reports the actual ImmedAqmMaxThreshold (MAXTH calculated per the MULPI Annex C, Annex N), rather than the configured value.

G.2.3.2.47 ImmedAqmRangeExponentRampFunction

This attribute specifies the range in nanoseconds of the ramp function used by the Immediate AQM algorithm and the Queue Protection algorithm. It is expressed as an exponent of 2, e.g., a value of 19 means the range of the ramp will be 2^19 = 524288 ns (roughly 524 μs).

G.2.3.2.48 DataRateUnitSetting

This attribute indicates the base unit for the Service Flow traffic rate attributes Maximum Sustained Traffic Rate (MaxTrafficRate), Minimum Reserved Traffic Rate (MinReservedRate), Peak Traffic Rate (PeakTrafficRate), and

Guaranteed Grant Rate (GuaranteedGrantRate). The value of this attribute allows for their interpretation in units of bps, kbps, Mbps, or Gbps. The default value for DataRateUnitSetting is bits per second (bps).

G.2.3.3 AggregateServiceFlow

This object describes the set of DOCSIS-QoS Aggregate Service Flows in a managed device.

References: [MULPIv4.0] Common Upstream and Downstream Quality-of-Service Parameter Encodings

Table 165 - AggregateServiceFlow Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
Id	UnsignedInt	Key			
Direction	IfDirection	R/O			
Priority	UnsignedByte	R/O			
MaxAggregateTrafficRate	BitRate	R/O			
MaxTrafficBurst	UnsignedInt	R/O			
MinReservedRate	BitRate	R/O			
MinReservedPkt	UnsignedShort	R/O		bytes	
PeakTrafficRate	BitRate	R/O			
DataRateUnitSetting	DataRateUnitType	R/O			
LowLatencyAsf	Boolean	R/O			
LowLatencySfid	UnsignedInt	R/O			
ClassicSfScn	AdminString	R/O			
LowLatencySfScn	AdminString	R/O			
AqmCouplingFactor	UnsignedByte	R/O	0..255	tenths	
SchedulingWeight	UnsignedByte	R/O			
QpEnable	EnumBits	R/O	queueProtectionEnabled(0), queueProtectionPerFlow(1) (deprecated)		
QpLatencyThreshold	UnsignedShort	R/O		microseconds	
QpQueuingScoreThreshold	UnsignedShort	R/O		microseconds	
QpDrainRateExponent	UnsignedByte	R/O		$\log_2$(bytes/second)	

G.2.3.3.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.3.3.2 Id

This key represents an identifier assigned to an Aggregate Service Flow by CMTS within a MAC Domain.

G.2.3.3.3 Direction

This attribute represents the direction of the Aggregate Service Flow.

G.2.3.3.4 Priority

This attribute represents the relative priority of the Aggregate Service Flow.

G.2.3.3.5 MaxAggregateTrafficRate

This attribute represents the 4-byte value of the maximum sustained traffic rate allowed for this Aggregate Service Flow. The value of the DataRateUnitSetting attribute defines the units of MaxAggregateTrafficRate.

G.2.3.3.6 MaxTrafficBurst

This attribute represents the token bucket size in bytes for this parameter set. The value is calculated from the byte following the MAC header HCS to the end of the CRC. This object is applied in conjunction with MaxTrafficRate to calculate maximum sustained traffic rate the token bucket size B (in bytes) for this Service Flow or an Aggregate Service Flow.

G.2.3.3.7 MinReservedRate

This attribute represents the 4-byte value of the guaranteed minimum rate allowed for this Aggregate Service Flow. The value of the DataRateUnitSetting attribute defines the units of MinReservedRate.

G.2.3.3.8 MinReservedPkt

This attribute represents Assumed Minimum Reserved Rate Packet Size for the Aggregate Service Flow.

G.2.3.3.9 PeakTrafficRate

This attribute represents the 4-byte value of the guaranteed minimum rate for this Aggregate Service Flow. The value of the DataRateUnitSetting attribute defines the units of PeakTrafficRate. A value of 0 means the peak traffic rate is not limited.

G.2.3.3.10 DataRateUnitSetting

This attribute indicates the base unit for the AggregateServiceFlow traffic rate attributes Maximum Aggregate Traffic Rate (MaxAggregateTrafficRate), Minimum Reserved Traffic Rate (MinReservedRate), and Peak Traffic Rate (PeakTrafficRate). The value of this attribute allows for their interpretation in units of bps, kbps, Mbps, or Gbps.

G.2.3.3.11 LowLatencyAsf

This attribute indicates whether this ASF is a Low Latency ASF.

G.2.3.3.12 LowLatencySfid

This attribute represents which of the constituent Service Flows within this Aggregate Service Flow is the Low Latency Service Flow.

G.2.3.3.13 ClassicSfScn

This attribute represents the Service Class Name that was used to instantiate the Classic Service Flow by the CMTS. This attribute may not be known by the CM, and could report an empty string.

G.2.3.3.14 LowLatencySfScn

This attribute represents the Service Class Name that was used to instantiate the Low Latency Service Flow by the CMTS. This attribute may not be known by the CM, and could report an empty string.

G.2.3.3.15 AqmCouplingFactor

This attribute represents the coupling factor for the AQMs between the Classic Service Flow and the Low Latency Service Flow.

G.2.3.3.16 SchedulingWeight

This attribute represents the value of this parameter defines the amount of scheduling weight the CMTS assigns to the Low Latency Service Flow of the ASF.

G.2.3.3.17 QpEnable

This attribute indicates the configured Queue Protection status of this Aggregated Service Flow. If the queueProtectionEnabled bit (bit 0) is set to '0', Queue Protection is disabled. If the queueProtectionEnabled bit is set to '1', Queue Protection is enabled. The 'queueProtectionPerFlow' bit (bit 1) is deprecated.

G.2.3.3.18 QpLatencyThreshold

This attribute represents the latency threshold for the Queue Protection function in the Latency Queue.

References: [MULPIv4.0] Queue Protection section.

G.2.3.3.19 QpQueuingScoreThreshold

This attribute represents Queuing Score Threshold for the Queue Protection function in the Latency Queue.

G.2.3.3.20 QpDrainRateExponent

This attribute represents the drain rate exponent for the Queue Protection function in the Low Latency Service Flow. The drain rate of the queuing score is expressed as an exponent of 2, in bytes/sec, e.g., a value of 17 means the Queue Protection function will use a value of: 2^17 bytes/sec.

G.2.3.4 ServiceFlow

The ServiceFlow object describes the set of DOCSIS-QoS Service Flows in a managed device.

References: [MULPIv4.0] Service Flows and Classifiers section.

Table 166 - ServiceFlow Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
Id	UnsignedInt	Key			
Sid	UnsignedShort	R/O	0..16383		
Direction	IfDirection	R/O			
Primary	Boolean	R/O			
ParamSetTypeStatus	EnumBits	R/O	active(0), admitted(1), provisioned(2)		
ChSetId	ChSetId	R/O			
AttrAssignSuccess	Boolean	R/O			
Dsid	Dsid	R/O			
MaxReqPerSidCluster	UnsignedByte	R/O		requests	
MaxOutstandingBytesPerSidCluster	UnsignedInt	R/O		bytes	
MaxTotBytesReqPerSidCluster	UnsignedInt	R/O		bytes	
MaxTimeInSidCluster	UnsignedShort	R/O		milliseconds	
BufferSize	UnsignedInt	R/O		bytes	
AggregateServiceFlowId	UnsignedInt	R/O			
Type	Enum	R/O	other(1), standalone(2), classic(3), lowLatency(4), nonLldHqos(5), reserved(6)		
AllowedAqBytes	UnsignedInt	R/O		bytes	

Table 167 - ServiceFlow Object Associations

Associated Object Name	Type	Near-end Multiplicity	Far-end Multiplicity	Label
ServiceFlowSidCluster	Association to ServiceFlowSidCluster	1	0..*	
PktClass	Association to PktClass	1	0..*	
ParamSet	Association to ParamSet	1	1..3	
AggregateServiceFlow	Association to AggregateServiceFlow	0..2	0..1	

G.2.3.4.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.3.4.2 Id

This key represents an identifier assigned to a Service Flow by CMTS within a MAC Domain. The value 0 is used only for the purpose of reporting instances of the PktClass object pertaining UDCs and not used for association of QoS classifiers to service flows.

References: [MULPIv4.0] Service Flow Identifier section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.3 Sid

Service Identifier (SID) assigned to an admitted or active Service Flow. This attribute reports a value of 0 if a Service ID is not associated with the Service Flow. Only active or admitted upstream Service Flows will have a Service ID (SID).

References: [MULPIv4.0] Service Identifier section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.4 Direction

This attribute represents the direction of the Service Flow.

G.2.3.4.5 Primary

This attribute reflects whether Service Flow is the primary or a secondary Service Flow.

G.2.3.4.6 ParamSetTypeStatus

This attribute represents the status of the service flow based on the admission state. 'active' bit set to '1' indicates that the service flow is active and that the corresponding QoS ParamSet is stored in the CMTS. 'admitted' bit set to '1' indicates that the service flow resources were reserved and that the corresponding QoS ParamSet is stored in the CMTS. 'provisioned' bit set to '1' indicates that the service flow was defined in the CM config file and that the corresponding QoS ParamSet is stored in the CMTS.

References: [MULPIv4.0] Service Flow Section.

G.2.3.4.7 ChSetId

This attribute represents the Channel Set Id associated with the service flow.

G.2.3.4.8 AttrAssignSuccess

If set to 'true', this attribute indicates that the current channel set associated with the service flow meets the Required and Forbidden Attribute Mask encodings. Since this attribute is not applicable for a CM, the CM always returns 'false'.

References: [MULPIv4.0] Service Flow section.

G.2.3.4.9 Dsid

This attribute indicates the DSID associated with the downstream service flow. downstream service flows without a DSID or upstream Service Flows report the value zero.

G.2.3.4.10 MaxReqPerSidCluster

This attribute specifies the maximum number of requests that a CM can make within a given SID Cluster before it needs to switch to a different SID Cluster to make further requests. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0.

References: [MULPIv4.0] Maximum Requests per SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.11 MaxOutstandingBytesPerSidCluster

This attribute specifies the maximum number of bytes for which a CM can have requests outstanding on a given SID Cluster. If defined number of bytes are outstanding and further requests are required, the CM needs to switch to a different SID Cluster if one is available. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0.

References: [MULPIv4.0] Maximum Outstanding Bytes per SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.12 MaxTotBytesReqPerSidCluster

This attribute specifies the maximum total number of bytes a CM can have requested using a given SID Cluster before it needs to switch to a different SID Cluster to make further requests. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0.

References: [MULPIv4.0] Maximum Total Bytes Requested per SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.13 MaxTimeInSidCluster

This attribute specifies the maximum time in milliseconds that a CM may use a particular SID Cluster before it needs to switch to a different SID Cluster to make further requests. A value of 0 indicates there is no limit. This attribute only applies to upstream Service Flows in 3.0 operation, in other cases it is reported as 0.

References: [MULPIv4.0] Maximum Time in the SID Cluster section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.14 BufferSize

This attribute indicates the buffer size for the service flow. For the CM this attribute only applies to upstream Service Flows, for the CMTS this attribute only applies to downstream Service Flows, in other cases it is reported as 0.

References: [MULPIv4.0] Buffer Control section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.4.15 AggregateServiceFlowId

This attribute indicates the Aggregate Service flow to which this Service Flow belongs. If this Service Flow is not part of an ASF, this attribute returns a '0'.

G.2.3.4.16 Type

This attribute reports the type of Service Flow.

'other' - Represents an undefined Service Flow type.

'standalone' - Represents a standalone Service Flow that is not associated with an ASF.

'classic' - Represents a classic Service Flow associated with a Low Latency ASF.

'lowLatency' - Represents a Low Latency Service Flow associated with a Low Latency ASF.

'nonLldHqos' - Represents a Service Flow that is a constituent of an Aggregate Service Flow (ASF) but when the ASF is not a Low Latency ASF.

'reserved' - Reserved value used by IPDR Service Definitions.

G.2.3.4.17 AllowedAqBytes

This attribute reports the estimate of the maximum number of bytes that are expected to be held in the upstream transmit queue simply because of the media access process (request/grant delay and PGS grant spacing), as calculated in the calcAllowedAQ function for each upstream Low Latency Service Flow. For other Service Flow types, the CM MAY report 0 for the AllowedAqBytes attribute.

References: [MULPIv3.1] allowed_AQ_bytes parameter calculated via the 'calcAllowedAQ()' function in Annex O.1 AQM Utility Functions

G.2.3.5 ServiceFlowSidCluster

This object defines the SID clusters associated with an upstream service flow.

References: [MULPIv4.0] Service Flow SID Cluster Assignments section in the Common Radio Frequency Interface Encodings Annex.

Table 168 - ServiceFlowSidCluster Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
ServiceFlowId	UnsignedInt	Key	1.. 4294967295		
Id	UnsignedByte	Key	0..7		
Ucid	ChId	Key	1..255		
Sid	UnsignedShort	R/O	1..16383		

G.2.3.5.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow SID cluster.

G.2.3.5.2 ServiceFlowId

This key represents the Service Flow ID for the service flow.

G.2.3.5.3 Id

This key represents the identifier of the SID Cluster.

References: [MULPIv4.0] SID Cluster ID section in the Common Radio Frequency Interface Encodings Annex.

G.2.3.5.4 Ucid

This key represents the upstream Channel ID mapped to the corresponding SID.

G.2.3.5.5 Sid

This attribute represents the SID assigned to the upstream channel in this SID Cluster.

G.2.4 QoS Statistics Objects

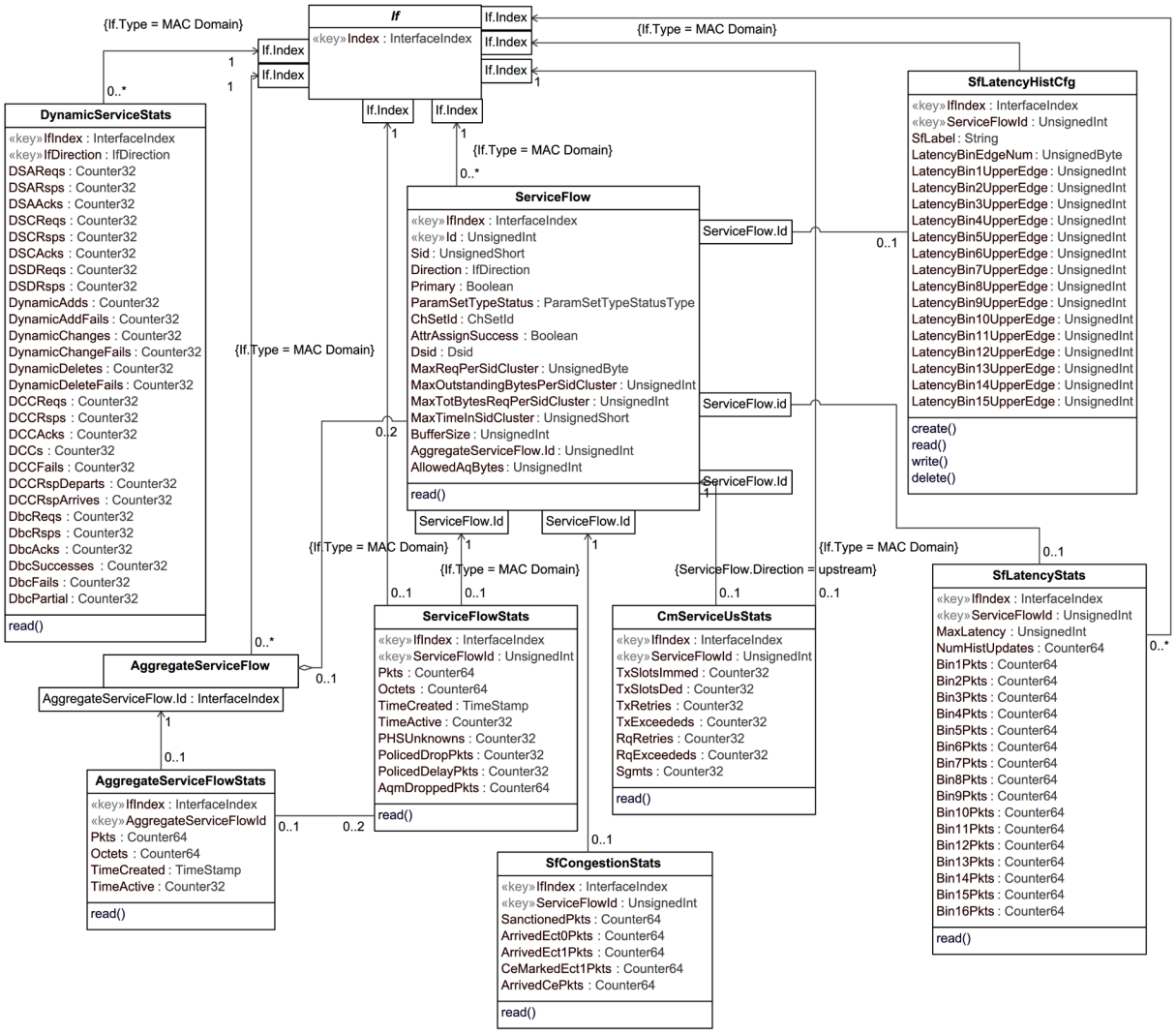


Figure 20 - QoS Statistics Information Model

G.2.4.1 ServiceFlowStats

This object describes statistics associated with the Service Flows in a managed device.

Table 169 - ServiceFlowStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
ServiceFlowId	UnsignedInt	Key	1..4294967295		
Pkts	Counter64	R/O		packets	
Octets	Counter64	R/O		bytes	
TimeCreated	TimeStamp	R/O			
TimeActive	Counter32	R/O		seconds	
PolicedDropPkts	Counter32	R/O		packets	
PolicedDelayPkts	Counter32	R/O		packets	
AqmDroppedPkts	Counter64	R/O		packets	

Attribute Name	Type	Access	Type Constraints	Units	Default
SanctionedPkts	Counter64	R/O		packets	

G.2.4.1.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.4.1.2 ServiceFlowId

This key represents an identifier assigned to a Service Flow by CMTS within a MAC Domain.

G.2.4.1.3 Pkts

For outgoing Service Flows, this attribute counts the number of Packet Data PDUs forwarded to this Service Flow. For incoming upstream CMTS service flows, this attribute counts the number of Packet Data PDUs actually received on the Service Flow identified by the SID for which the packet was scheduled. CMs not classifying downstream packets may report this attribute's value as 0 for downstream Service Flows. This attribute does not count MAC-specific management messages. Particularly for UGS flows, packets sent on the primary Service Flow in violation of the UGS grant size should be counted only by the instance of this attribute that is associated with the primary service flow. Unclassified upstream user data packets (i.e., non- MAC-management) forwarded to the primary upstream Service Flow should be counted by the instance of this attribute that is associated with the primary service flow. This attribute does include packets counted by ServiceFlowPolicedDelayPkts, but does not include packets counted by ServiceFlowPolicedDropPkts. This counter's last discontinuity is the ifCounterDiscontinuityTime for of the associated MAC Domain interface index.

G.2.4.1.4 Octets

This attribute indicates the count of the number of octets from the byte after the MAC header HCS to the end of the CRC for all packets counted in the ServiceFlowPkts attribute for this row. Note that this counts the octets after payload header suppression and before payload header expansion have been applied. This counter's last discontinuity is the ifCounterDiscontinuityTime for of the associated MAC Domain interface index.

G.2.4.1.5 TimeCreated

This attribute indicates the value of sysUpTime when the service flow was created.

G.2.4.1.6 TimeActive

This attribute indicates the number of seconds that the service flow has been active. This counter's last discontinuity is the ifCounterDiscontinuityTime for of the associated MAC Domain interface index.

G.2.4.1.7 PolicedDropPkts

For upstream service flows, this attribute counts the number of Packet Data PDUs classified to this service flow dropped due to: (1) exceeding the selected Buffer Size for the service flow (see the Buffer Control section in the Common Radio Frequency Interface Encodings Annex of [MULPIv4.0]; or (2) UGS packets dropped due to exceeding the Unsolicited Grant Size with a Request/Transmission policy that requires such packets to be dropped. Classified packets dropped due to other reasons need to be counted in either AqmDroppedPkts or ifOutDiscards for the interface of this service flow (depending on the reason for the discard). This attribute reports 0 for downstream service flows. This counter's last discontinuity is the ifCounterDiscontinuityTime of the associated MAC Domain interface index.

G.2.4.1.8 PolicedDelayPkts

This attribute counts only outgoing packets delayed in order to maintain the Maximum Sustained Traffic Rate. This attribute will always report a value of 0 for UGS flows because the Maximum Sustained Traffic Rate does not apply. This attribute is 0 for incoming service flows. This counter's last discontinuity is the ifCounterDiscontinuityTime of the associated MAC Domain interface index.

G.2.4.1.9 AqmDroppedPkts

For upstream service flows on which AQM is enabled, this attribute reports the count of the number of Packet Data PDUs classified to this service flow dropped due to Active Queue Management drop decisions. Classified packets dropped due to other reasons are counted in either PolicedDropPkts or ifOutDiscards for the interface of this service flow (depending on the reason for the discard). The cable modem reports zero for this attribute for downstream service flows.

References: [MULPIv4.0] Active Queue Management Algorithm Section.

G.2.4.1.10 Sanctioned Packets

This attribute represents the number of packets redirected to the Classic Service Flow queue by the Queue Protection function.

G.2.4.2 AggregateServiceFlowStats

This object describes statistics associated with the Aggregate Service Flows in a managed device.

Table 170 - AggregateServiceFlowStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
AggregateServiceFlowId	UnsignedInt	Key	1..4294967295		
Pkts	Counter64	R/O		packets	
Octets	Counter64	R/O		bytes	
TimeCreated	TimeStamp	R/O			
TimeActive	Counter32	R/O		seconds	

G.2.4.2.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.4.2.2 AggregateServiceFlowId

This key represents an identifier assigned to an Aggregate Service Flow by CMTS within a MAC Domain.

G.2.4.2.3 Pkts

This attribute provides the sum of the Packet Data PDUs forwarded on the Low Latency and Classic Service Flows aggregated within this ASF. This counter's last discontinuity is the ifCounterDiscontinuityTime for of the associated MAC Domain interface index.

G.2.4.2.4 Octets

This attribute represents the sum of the number of octets for the Low Latency and Classic Service Flows aggregated within this ASF. For each packet, the count begins from the byte after the MAC header HCS to the end of the CRC.

G.2.4.2.5 TimeCreated

This attribute indicates the value of sysUpTime when the aggregate service flow was created.

G.2.4.2.6 TimeActive

This attribute indicates the number of seconds that the aggregate service flow has been active. This counter's last discontinuity is the ifCounterDiscontinuityTime for of the associated MAC Domain interface index.

G.2.4.3 DynamicServiceStats

This object describes statistics associated with the Dynamic Service Flows, Dynamic Channel Changes and Dynamic Bonding Changes in a managed device within a MAC Domain. For each MAC Domain there are two instances for the for the upstream and downstream direction. On the CMTS, the downstream direction instance indicates messages transmitted or transactions originated by the CMTS. The upstream direction instance indicates messages received or transaction originated by the CM. On the CM, the downstream direction instance indicates messages received or transactions originated by the CMTS. The upstream direction instance indicates messages transmitted by the CM or transactions originated by the CM.

Table 171 - DynamicServiceStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
IfDirection	IfDirection	Key			
DSAReqs	Counter32	R/O		messages	
DSARsps	Counter32	R/O		messages	
DSAAcks	Counter32	R/O		messages	
DSCReqs	Counter32	R/O		messages	
DSCRsps	Counter32	R/O		messages	
DSCAcks	Counter32	R/O		messages	
DSDReqs	Counter32	R/O		messages	
DSDRsps	Counter32	R/O		messages	
DynamicAdds	Counter32	R/O		messages	
DynamicAddFails	Counter32	R/O		messages	
DynamicChanges	Counter32	R/O		messages	
DynamicChangeFails	Counter32	R/O		messages	
DynamicDeletes	Counter32	R/O		messages	
DynamicDeleteFails	Counter32	R/O		messages	
DCCReqs	Counter32	R/O		messages	
DCCRsps	Counter32	R/O		messages	
DCCAcks	Counter32	R/O		messages	
DCCs	Counter32	R/O		messages	
DCCFails	Counter32	R/O		messages	
DCCRspDeparts	Counter32	R/O		messages	
DCCRspArrives	Counter32	R/O		messages	
DbcReqs	Counter32	R/O		messages	
DbcRsps	Counter32	R/O		messages	
DbcAcks	Counter32	R/O		messages	
DbcSuccesses	Counter32	R/O		transactions	
DbcFails	Counter32	R/O		transactions	
DbcPartial	Counter32	R/O		transactions	

G.2.4.3.1 IfIndex

This key represents the interface index of the MAC Domain.

G.2.4.3.2 IfDirection

This attribute indicates the interface direction for the instance the statistics are collected.

G.2.4.3.3 DSAReqs

This attribute indicates the number of Dynamic Service Addition Requests, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Addition section; [RFC 2863]

G.2.4.3.4 DSARsps

The number of Dynamic Service Addition Responses, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Addition section; [RFC 2863]

G.2.4.3.5 DSAAcks

The number of Dynamic Service Addition Acknowledgements, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Addition section; [RFC 2863]

G.2.4.3.6 DSCReqs

The number of Dynamic Service Change Requests, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Change section; [RFC 2863]

G.2.4.3.7 DSCRsps

The number of Dynamic Service Change Responses, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Change section; [RFC 2863]

G.2.4.3.8 DSCAcks

The number of Dynamic Service Change Acknowledgements, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Change section; [RFC 2863]

G.2.4.3.9 DSDReqs

The number of Dynamic Service Delete Requests, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Deletion section; [RFC 2863]

G.2.4.3.10 DSDRsps

The number of Dynamic Service Delete Responses, including retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Change section; [RFC 2863]

G.2.4.3.11 DynamicAdds

The number of successful Dynamic Service Addition transactions. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Addition section; [RFC 2863]

G.2.4.3.12 DynamicAddFails

The number of failed Dynamic Service Addition transactions. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Addition section; [RFC 2863]

G.2.4.3.13 DynamicChanges

The number of successful Dynamic Service Change transactions. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Change section; [RFC 2863]

G.2.4.3.14 DynamicChangeFails

The number of failed Dynamic Service Change transactions. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Change section; [RFC 2863]

G.2.4.3.15 DynamicDeletes

The number of successful Dynamic Service Delete transactions. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Delete section; [RFC 2863]

G.2.4.3.16 DynamicDeleteFails

The number of failed Dynamic Service Delete transactions. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Service Delete section; [RFC 2863]

G.2.4.3.17 DCCReqs

The number of Dynamic Channel Change Request messages traversing an interface. This count is nonzero only on downstream direction rows. This count should include the number of retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.18 DCCRsps

The number of Dynamic Channel Change Response messages traversing an interface. This count is nonzero only on upstream direction rows. This count should include the number of retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.19 DCCAcks

The number of Dynamic Channel Change Acknowledgement messages traversing an interface. This count is nonzero only on downstream direction rows. This count should include the number of retries. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.20 DCCs

The number of successful Dynamic Channel Change transactions. This count is nonzero only on downstream direction rows. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.21 DCCFails

The number of failed Dynamic Channel Change transactions. This count is nonzero only on downstream direction rows. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.22 DccRspDeparts

This attribute contains the number of Dynamic Channel Change Response (depart) messages. It only applies to upstream direction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.23 DccRspArrives

This attribute contains the number of Dynamic Channel Change Response (arrive) messages and should include retries. It only applies to the upstream direction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Downstream and/or Upstream Channel Changes section; [RFC 2863]

G.2.4.3.24 DbcReqs

This attribute contains the number of Dynamic Bonding Change Requests, including retries. It only applies to the upstream direction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Bonding Change (DBC) section; [RFC 2863]

G.2.4.3.25 DbcRsps

This attribute contains the number of Dynamic Bonding Change Responses, including retries. It only applies to the upstream direction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Bonding Change (DBC) section; [RFC 2863]

G.2.4.3.26 DbcAcks

This attribute contains the number of Dynamic Bonding Change Acknowledgements, including retries. It only applies to the downstream direction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Bonding Change (DBC) section; [RFC 2863]

G.2.4.3.27 DbcSuccesses

This attribute contains the number of fully successful Dynamic Bonding Change transactions. It only applies to the downstream direction and does not include DBC transactions that result in Partial Service. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Bonding Change (DBC) section; [RFC 2863]

G.2.4.3.28 DbcFails

This attribute contains the number of failed Dynamic Bonding Change transactions. It only applies to the downstream direction. Note that Partial Service is not considered a failed transaction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Bonding Change (DBC) section; [RFC 2863]

G.2.4.3.29 DbcPartial

This attribute contains the number of unsuccessful Dynamic Bonding Change transactions that result in Partial Service. IT only applies to the downstream direction. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Dynamic Bonding Change (DBC) section; [RFC 2863]

G.2.4.4 CmServiceUsStats

This object defines DOCSIS MAC services primitive statistics of upstream service flows.

References: [MULPIv4.0] Upstream Data Transmission section.

Table 172 - CmServiceUsStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
ServiceFlowId	UnsignedInt	Key	1.. 4294967295		
TxSlotsImmed	Counter32	R/O		minislots	
TxSlotsDed	Counter32	R/O		minislots	
TxRetries	Counter32	R/O		attempts	
TxExceededs	Counter32	R/O		attempts	
RqRetries	Counter32	R/O		attempts	
RqExceededs	Counter32	R/O		attempts	
Sgmts	Counter32	R/O		segments	

G.2.4.4.1 IfIndex

This key represents the interface index of the MAC Domain to which this instance applies.

G.2.4.4.2 ServiceFlowId

This key represents the Service Flow ID for the service flow.

References: [MULPIv4.0] QoS section

G.2.4.4.3 TxSlotsImmed

This attribute contains the number of upstream minislots which have been used to transmit data PDUs in immediate (contention) mode. This includes only those PDUs that are presumed to have arrived at the headend (i.e., those which were explicitly acknowledged.) It does not include retransmission attempts or minislots used by Requests. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream Bandwidth Allocation section; [RFC 2863]

G.2.4.4.4 TxSlotsDed

This attribute contains the number of upstream minislots which have been used to transmit data PDUs in dedicated mode (i.e., as a result of a unicast Data Grant). Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream Data Transmission section; [RFC 2863]

G.2.4.4.5 TxRetries

This attribute contains the number of attempts to transmit data PDUs containing requests for acknowledgment that did not result in acknowledgment. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime for the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream Bandwidth Allocation section; [RFC 2863]

G.2.4.4.6 TxExceededs

This attribute contains the number of data PDUs transmission failures due to excessive retries without acknowledgment. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream Bandwidth Allocation section; [RFC 2863]

G.2.4.4.7 RqRetries

This attribute contains the number of attempts to transmit bandwidth requests which did not result in acknowledgment. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream Bandwidth Allocation section; [RFC 2863].

G.2.4.4.8 RqExceededs

This attribute contains the number of requests for bandwidth which failed due to excessive retries without acknowledgment. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream Bandwidth Allocation section; [RFC 2863]

G.2.4.4.9 Sgmts

This attribute contains the number of segments transmitted on this service flow. Discontinuities in the value of this counter can occur at reinitialization of the managed system, and at other times as indicated by the value of ifCounterDiscontinuityTime of the associated MAC Domain interface index.

References: [MULPIv4.0] Upstream and Downstream Common Aspects section; [RFC 2863]

G.2.4.5 SfLatencyHistCfg

This object defines the latency histogram structure for each upstream histogram enabled service flow. The CM MUST create an active instance of this object for every upstream service flow for which histogram calculation is enabled via the Latency Histogram Encodings TLV. The activation of an instance of this object enables histogram calculation for the indexed service flow. The deactivation or deletion of an instance of this object disables histogram calculation for the indexed service flow. The CM MUST delete an instance of this object if the indexed service flow is deleted (e.g., because of a DSD operation) or the histogram calculation is disabled via the Latency Histogram Encodings TLV (i.e., because of a DSC operation).

Prior to deactivating or deleting an instance of the SfLatencyHistCfg object, the CM MUST, for the indexed service flow, conclude any Latency Reporting Snapshot in progress (see Section D.3), and delete the corresponding instance in the SfLatencyStats object. Once the SfLatencyHistCfg object is deactivated or deleted the CM does not collect any further latency statistics. The CM MUST NOT persist instances created in the SfLatencyHistCfg object across reinitializations.

Bins are, by definition, consecutively numbered from one to sixteen. At least one and at most fifteen upper edges can be configured. Each configured upper edge is also the lower edge for the next bin (e.g., Bin 5 lower edge is defined by *LatencyBin4UpperEdge*). The lower edge of Bin 1 is defined as zero, and the upper edge of the final active bin (i.e., the bin corresponding to *LatencyBinEdgeNum* + 1) is defined as infinity. The CM validates the configured bin edges prior to activation.

The Service Flow Latency Estimate values corresponding to Bin N are bounded as follows: Bin N lower edge ≤ Service Flow *Latency Estimate* < Bin N upper edge.

References: See [MULPIv4.0] Latency Histogram Calculation section.

Table 173 - SfLatencyHistCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface index of MAC Domain Interface	N/A	
ServiceFlowId	UnsignedInt	Key	1.. 4294967295		
SfLabel	String	R/W	SIZE (0..15)		'unknown'
LatencyBinEdgeNum	UnsignedByte	R/W	0..15		0
LatencyBin1UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin2UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin3UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin4UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin5UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin6UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin7UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin8UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin9UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin10UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin11UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin12UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin13UpperEdge	UnsignedInt	R/W		microseconds	0
LatencyBin14UpperEdge	UnsignedInt	R/W		microseconds	0

Attribute Name	Type	Access	Type Constraints	Units	Default
LatencyBin15UpperEdge	UnsignedInt	R/W		microseconds	0

G.2.4.5.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.4.5.2 ServiceFlowId

This key represents the Service Flow enabled to calculate queue latency histograms.

G.2.4.5.3 SfLabel

This attribute represents the Service Class Name or a Label (configured by the operator) to be used in Service Flow Latency Performance Reporting (see Section D.3). The CM SHOULD default this attribute to the SCN if it's known. If the SCN is not known, the CM MUST default this attribute to the value "unknown". The Operator can set this attribute to any string.

G.2.4.5.4 LatencyBinEdgeNum

This attribute represents the number of consecutive configured upper bin edges. The number of histogram bins is then *LatencyBinEdgeNum* + 1.

G.2.4.5.5 LatencyBin1UpperEdge to LatencyBin15UpperEdge

These attributes represent the upper edges of each bin, bins 1 through 15. The histogram calculation behavior is undefined if configured upper bin edges are not monotonically increasing beginning with the LatencyBin1UpperEdge (e.g., if for bin number 'n', LatencyBin(n)UpperEdge ≥ LatencyBin(n+1)UpperEdge).

G.2.4.6 SfLatencyStats

This object contains the latency statistics calculated by the CM for each histogram enabled upstream service flow. The SfLatencyStats object entry describes latency statistics (histogram of latencies, Max Latency, Sanctioned packets, etc.) at the CM for the upstream service flow. The CM MUST create an instance for each ServiceFlowId value for which histogram calculation is enabled. The CM MUST delete the instance when histogram calculation is disabled for the service flow.

The CM MUST NOT delete SfLatencyStats instances while the corresponding SfLatencyHistCfg instance is active. The CM MUST NOT persist instances created in the SfLatencyStats object across reinitializations.

References: [MULPIv4.0]

Table 174 - SfLatencyStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface index of MAC Domain Interface		
ServiceFlowId	UnsignedInt	Key	1.. 4294967295		
MaxLatency	UnsignedInt	R/O		microseconds	-
NumHistUpdates	Counter64	R/O		-	-
Bin1PktCount	Counter64	R/O		packets	
Bin2PktCount	Counter64	R/O		packets	
Bin3PktCount	Counter64	R/O		packets	
Bin4PktCount	Counter64	R/O		packets	
Bin5PktCount	Counter64	R/O		packets	
Bin6PktCount	Counter64	R/O		packets	

Attribute Name	Type	Access	Type Constraints	Units	Default
Bin7PktCount	Counter64	R/O		packets	-
Bin8PktCount	Counter64	R/O		packets	-
Bin9PktCount	Counter64	R/O		packets	-
Bin10PktCount	Counter64	R/O		packets	-
Bin11PktCount	Counter64	R/O		packets	-
Bin12PktCount	Counter64	R/O		packets	-
Bin13PktCount	Counter64	R/O		packets	-
Bin14PktCount	Counter64	R/O		packets	-
Bin15PktCount	Counter64	R/O		packets	-
Bin16PktCount	Counter64	R/O		packets	-

G.2.4.6.1 IfIndex

This key represents the interface index of the MAC Domain of the Service Flow.

G.2.4.6.2 ServiceFlowId

This key represents the Service Flow to which this instance applies.

G.2.4.6.3 MaxLatency

This attribute represents maximum latency observed.

G.2.4.6.4 NumHistUpdates

This attribute represents count of updates to the queue latency histogram. In cases where a latency estimate is not generated for every packet (e.g., subsampling in the case of Low Latency Service Flow or estimates every update interval in the case of Classic Service Flow) this attribute provides information regarding the fidelity of the histogram bin counts. In the case of a Classic Service Flow, this counter will only increment if packets have been enqueued during the last update interval.

G.2.4.6.5 Bin1PktCount to Bin16PktCount

These attributes represent the count of packets whose latency falls within each histogram Bin. If the number of bin edges configured is 'n' then only the first 'n+1' bin counts are valid. The CM MUST report zero for Bin'*X*'PktCount for values of '*X*' greater than n+1.

G.2.4.7 SfCongestionStats

The SfCongestionStats object defines congestion statistics for CM upstream service flows The CM MUST create an active instance of SfCongestionStats for every upstream Low Latency service flow. The CM MAY create an active instance of SfCongestionStats for other upstream service flow types. The CM MUST delete an instance of SfCongestionStats if the indexed service flow is deleted.

Table 175 - SfCongestionStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface index of the MAC Domain interface		
ServiceFlowId	UnsignedInt	Key	1.. 4294967295		
SanctionedPkts	Counter64	R/O		packets	
ArrivedEct0Pkts	Counter64	R/O		packets	
ArrivedEct1Pkts	Counter64	R/O		packets	
CeMarkedEct1Pkts	Counter64	R/O		packets	
ArrivedCePkts	Counter64	R/O		packets	

G.2.4.7.1 IfIndex

This key represents the interface index of the MAC Domain where the upstream Service Flow has been created.

G.2.4.7.2 ServiceFlowId

This key represents the upstream Service Flow to which this instance applies.

G.2.4.7.3 SanctionedPkts

This attribute counts the number of packets redirected from the Low Latency Service Flow to the Classic Service Flow. For other Service Flow types in the CM, this counter reports 0.

G.2.4.7.4 ArrivedEct0Pkts

This attribute reports the count of packets that arrived marked as ECT0. This attribute includes only those packets classified into the Service Flow, not those sanctioned into the Service Flow.

G.2.4.7.5 ArrivedEct1Pkts

This attribute reports the count of packets that arrived marked as ECT1. This attribute includes only those packets classified into the Service Flow, not those sanctioned into the Service Flow.

G.2.4.7.6 CeMarkedEct1Pkts

This attribute reports the count of packets that arrived marked as ECT1 and were re-marked as Congestion Experienced (CE) by the CM. This attribute includes only those packets classified into the Service Flow, not those sanctioned into the Service Flow.

G.2.4.7.7 ArrivedCePkts

This attribute reports the count of packets that arrived marked as Congestion Experienced (CE). This attribute includes only those packets classified into the Service Flow, not those sanctioned into the Service Flow.

G.2.5 DSID Objects

This section defines Downstream Service Identifier (DSID) related objects.

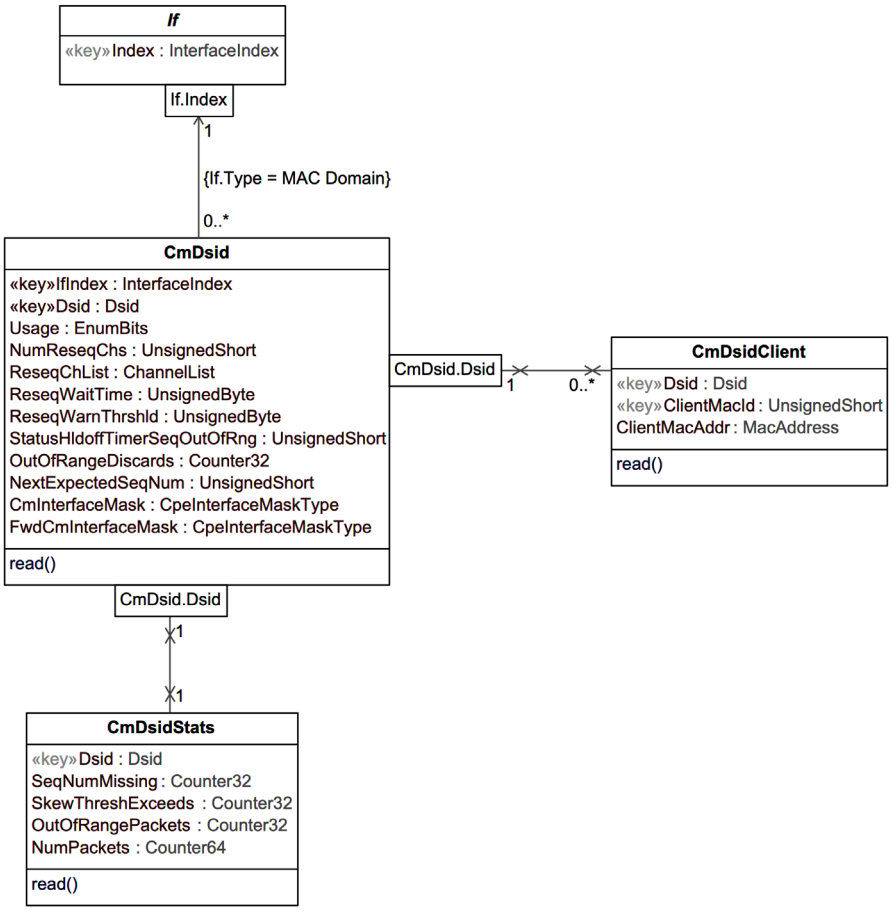


Figure 21 - DSID Information Model

G.2.5.1 CmDsid

This object describes the DSID information stored in the CM.

The CM reports the current status of existing DSIDs. When a DSID is created during the registration process or a DBC transaction, a corresponding object instance is created. If a DSID is deleted or changed via a DBC message the corresponding object instance is deleted or updated respectively.

Table 176 - CmDsid Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key	Interface Index of MAC Domain interface		
Dsid	Dsid	Key			
Usage	EnumBits	R/O	resequencing(0) multicastCapable(1)		
NumReseqChs	UnsignedShort	R/O	0 | 1..65535		
ReseqChList	ChannelList	R/O	SIZE (0|2..255)		
ReseqWaitTime	UnsignedByte	R/O	0 | 1..180	hundredMicroseconds	
ReseqWarnThrshId	UnsignedByte	R/O	0..179	hundredMicroseconds	
StatusHldoffTimerSeqOutOfRng	UnsignedShort	R/O		20 milliseconds	
OutOfRangeDiscards	Counter32	R/O			

Attribute Name	Type	Access	Type Constraints	Units	Default
NextExpectedSeqNum	UnsignedShort	R/O			
CmInterfaceMask	CpeInterfaceMaskType	R/O			
FwdCmInterfaceMask	CpeInterfaceMaskType	R/O			

G.2.5.1.1 IfIndex

This key represents the interface index of the MAC Domain associated with the DSID.

G.2.5.1.2 Dsid

This key represents the DSID.

G.2.5.1.3 Usage

This attribute indicates the properties of the DSID. The bits are defined as follows:

- 'resequencing'
  This bit is set to 1 for a Resequencing DSID.
- 'multicastCapable'
  This bit is set to 1 for a DSID that is capable of transporting multicast traffic (e.g., the DSID has multicast forwarding attributes).

G.2.5.1.4 NumReseqChs

This attribute represents the number of channels in the downstream resequencing channel list for this DSID. When a DSID is used only for a non-bonded multicast replication, this object returns a value of 0.

G.2.5.1.5 ReseqChList

This attribute represents the Downstream Channel Set over which the DSID is being resequenced.

G.2.5.1.6 ReseqWaitTime

This attribute represents the DSID Resequencing Wait Time that is used for this DSID. This attribute is only valid when the Usage attribute has the resequencing bit set to 1. This attribute returns a value of 0 when the Usage attribute has the resequencing bit set to 0.

G.2.5.1.7 ReseqWarnThrshld

This attribute represents the DSID Resequencing Warning Threshold that is used for this DSID. The value of 0 indicates that the threshold warnings are disabled. This attribute is only valid when the Usage attribute has the resequencing bit set to 1. This attribute returns a value of 0 when the Usage attribute has the resequencing bit set to 0.

G.2.5.1.8 StatusHldoffTimerSeqOutOfRng

This attribute represents the hold-off timer for reporting Out-of-Range Events via the CM-STATUS MAC Management message. This attribute is only valid when the Usage attribute has the resequencing bit set to 1. This attribute returns a value of 0 when the Usage attribute has the resequencing bit set to 0.

G.2.5.1.9 OutOfRangeDiscards

This attribute represents the current count of out-of-range packets discarded by the CM for a given resequencing context since an in-range packet was received. When this count exceeds 1000 and more than two minutes have elapsed since an in-range packet was received, the CM will reacquire sequence numbers for this resequencing context.

G.2.5.1.10 NextExpectedSeqNum

This attribute represents the Next Expected Packet Sequence Number for a given resequencing context. This attribute is only valid when the Usage attribute has the resequencing bit set to 1. This attribute returns a value of 0 when the Usage attribute has the resequencing bit set to 0.

G.2.5.1.11 CmInterfaceMask

This attribute represents the bitmap of the interfaces communicated to the CM in a Multicast DSID encoding.

G.2.5.1.12 FwdCmInterfaceMask

This attribute represents the bitmap of the interfaces to which the CM forwards multicast traffic: a logical OR of interfaces identified in CmInterfaceMask and interfaces associated with the client MAC addresses identified in the instances for this DSID.

G.2.5.2 CmDsidStats

This object defines a set of statistics the CM collects per DSID.

Table 177 - CmDsidStats Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
IfIndex	InterfaceIndex	Key			
Dsid	Dsid	Key			
SeqNumMissing	Counter32	R/O			
SkewThreshExceeds	Counter32	R/O		packets	
OutOfRangePackets	Counter32	R/O		packets	
NumPackets	Counter64	R/O		packets	

G.2.5.2.1 IfIndex

This key represents the interface index of the MAC Domain associated with the DSID.

G.2.5.2.2 Dsid

This key represents the DSID.

G.2.5.2.3 SeqNumMissing

This attribute counts the number of times the Next Expected Packet Sequence Number is declared lost. In this case one or more data packets are lost. This is generally caused by downstream packet loss.

References: [MULPIv4.0] Downstream Sequencing section

G.2.5.2.4 SkewThreshExceeds

This attribute counts in-range sequenced packets which were successfully received by the CM after a wait time longer than the Resequencing Warning Threshold.

References: [MULPIv4.0] Downstream Sequencing section

G.2.5.2.5 OutOfRangePackets

This attribute counts the number of packets Counter received in a DSID reassembly context where the sequence number which is out of range.

References: [MULPIv4.0] Receive Channels section

G.2.5.2.6 NumPackets

This attribute counts the total number of data packets of a DSID context forwarded for further processing.

G.2.5.3 CmDsidClient

This object contains the client MAC addresses that the CMTS requests that the CM uses to replicate Multicast DSIDs during registration or during a DBC transaction.

When a DSID is created that includes client MAC addresses, or when client MAC addresses are added to a DSID, new rows are created to indicate the added client MAC addresses. When a Client MAC address is deleted from a DSID, the corresponding row is deleted. When a DSID is deleted, all corresponding rows are deleted, too.

References: [MULPIv4.0] DSID Encodings section in the Common Radio Frequency Interface Encodings Annex.

Table 178 - CmDsidClient Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Dsid	Dsid	Key			
MacId	UnsignedShort	Key	1..65535		
MacAddr	MacAddress	R/O			

G.2.5.3.1 Dsid

This key defines the DSID with which the client MAC addresses are associated.

G.2.5.3.2 MacId

This key defines a uniquely identified client MAC address associated with the DSID.

G.2.5.3.3 MacAddr

This attribute defines a client MAC address to which multicast traffic labeled with this DSID should be forwarded.

G.2.6 CM Provisioning Objects

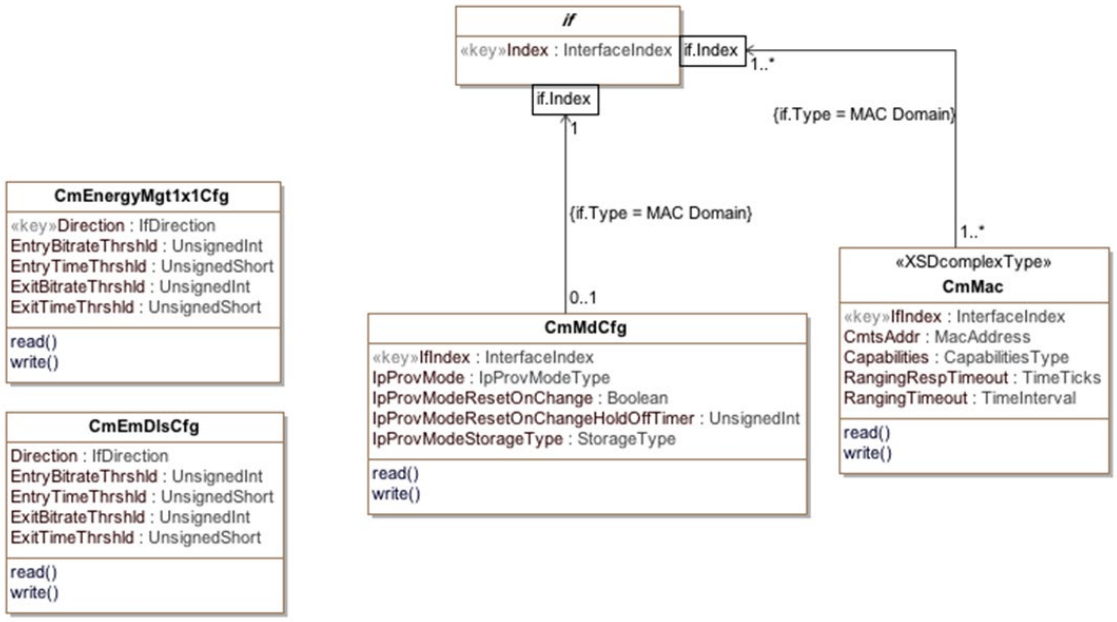


Figure 22 - CM MAC Domain Configuration Information Model

G.2.6.1 CmMdCfg

This object contains MAC domain level control and configuration attributes for the CM.

References: [MULPIv4.0] IP Provisioning Mode Override section.

Table 179 - CmMdCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
ifIndex	InterfaceIndex	Key	InterfaceIndex of MAC Domain interface		
IpProvMode	Enum	R/W	ipv4Only(0) ipv6Only(1) honorMdd(4)		honorMdd
IpProvModeResetOnChange	TruthValue	R/W	true(1) false(2)		false
IpProvModeResetOnChangeHoldOffTimer	UnsignedInt	R/W	0...300	seconds	0
IpProvModeStorageType	StorageType	R/W	volatile(2) nonVolatile(3)		nonVolatile

G.2.6.1.1 ifIndex

This key represents the interface index of the MAC Domain to which this instance applies.

G.2.6.1.2 IpProvMode

This attribute specifies whether the CM honors or ignores the CMTS MDD TLV 5.1 setting in order to configure its IP provisioning mode. The CM relies upon the CMTS to facilitate the successful IP address acquisition independently of the MDD.

When this attribute is set to 'ipv4Only' the CM will initiate the acquisition of a single IPv4 address for the CM management stack.

When this attribute is set to 'ipv6Only' the CM will initiate the acquisition of a single IPv6 address for the CM management stack.

When this attribute is set to 'honorMdd', the CM will initiate the acquisition of an IP address as directed by the MDD message sent by the CMTS.

References: [MULPIv4.0] IP Initialization Parameters TLV section

G.2.6.1.3 IpProvModeResetOnChange

This attribute determines whether the CM is to automatically reset upon a change to the IpProvMode attribute. The IpProvModeResetOnChange attribute has a default value of 'false' which means that the CM does not reset upon change to IpProvMode attribute. When this attribute is set to 'true', the CM resets upon a change to the IpProvMode attribute.

References: [MULPIv4.0] IP Initialization Parameters TLV section

G.2.6.1.4 IpProvModeResetOnChangeHoldOffTimer

This attribute determines how long a CM with IpProvModeResetOnChange set to 'true' waits to reset. When the IpProvModeResetOnChange attribute is set to 'true', the CM will decrement from the configured timer value before resetting. The default value of the IpProvModeResetOnChangeHoldOffTimer is 0 seconds which is equivalent to an immediate reset.

References: [MULPIv4.0] IP Initialization Parameters TLV section

G.2.6.1.5 IpProvModeStorageType

This attribute determines if the CM persists the value of IpProvMode across a single reset or across all resets. The default value of IpProvModeStorageType is 'nonVolatile' which means that the CM persists the value of IpProvMode

across all resets. The CM persists the value of IpProvMode across only a single reset when IpProvModeStorageType is set to 'volatile'.

References: [MULPIv4.0] IP Initialization Parameters TLV section

G.2.6.2 CmEnergyMgt1x1Cfg

This object provides configuration state information on the CM for the Energy Management 1x1 Mode feature.

The values of these attributes are not persisted across reinitialization.

Reference: [MULPIv4.0] Energy Management 1x1 Mode Encodings section.

Table 180 - CmEnergyMgt1x1Cfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Direction	IfDirection	Key			
EntryBitrateThrshld	UnsignedInt	R/W		bps	Vendor-specific
EntryTimeThrshld	UnsignedShort	R/W	1..65535	seconds	Vendor-specific
ExitBitrateThrshld	UnsignedInt	R/W		bps	Vendor-specific
ExitTimeThrshld	UnsignedShort	R/W	1..65535	seconds	Vendor-specific

G.2.6.2.1 Direction

This key attribute indicates whether the threshold applies to the upstream or downstream.

G.2.6.2.2 EntryBitrateThrshld

This attribute specifies the upstream or downstream bitrate threshold (in bps) below which the CM will request to enter Energy Management 1x1 Mode operation.

G.2.6.2.3 EntryTimeThrshld

This attribute specifies the number of consecutive seconds that the upstream or downstream data rate needs to remain below the Upstream or Downstream Entry Bitrate Threshold in order to determine that a transition to Energy Management 1x1 Mode is required.

G.2.6.2.4 ExitBitrateThrshld

This attribute specifies the upstream or downstream bitrate threshold (in bps) above which the CM will request to leave Energy Management 1x1 Mode operation.

G.2.6.2.5 ExitTimeThrshld

This attribute specifies the number of consecutive seconds that the upstream or downstream data rate needs to remain above the Upstream or Downstream Exit Bitrate Threshold in order to determine that a transition out of Energy Management 1x1 Mode is required.

G.2.6.3 CmEmDlsCfg

This object provides configuration state information on the CM for the Energy Management DLS Mode feature.

The values of these attributes are not persisted across reinitialization.

Reference: [MULPIv4.0] Energy Management DLS Mode Encodings section.

Table 181 - CmEmDlsCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Direction	IfDirection	Key			
EntryBitrateThrshld	UnsignedInt	R/W		bps	Vendor-specific

Attribute Name	Type	Access	Type Constraints	Units	Default
EntryTimeThrshld	UnsignedShort	R/W	1..65535	seconds	Vendor-specific
ExitBitrateThrshld	UnsignedInt	R/W		bps	Vendor-specific
ExitTimeThrshld	UnsignedShort	R/W	1..65535	seconds	Vendor-specific

G.2.6.3.1 Direction

This key attribute indicates whether the threshold applies to the upstream or downstream.

G.2.6.3.2 EntryBitrateThrshld

This attribute specifies the upstream or downstream bitrate threshold (in bps) below which the CM will request to enter Energy Management DLS Mode operation.

G.2.6.3.3 EntryTimeThrshld

This attribute specifies the number of consecutive seconds that the upstream or downstream data rate needs to remain below the Upstream or Downstream Entry Bitrate Threshold in order to determine that a transition to Energy Management DLS Mode is required.

G.2.6.3.4 ExitBitrateThrshld

This attribute specifies the upstream or downstream bitrate threshold (in bps) above which the CM will request to leave Energy Management DLS Mode operation.

G.2.6.3.5 ExitTimeThrshld

This attribute specifies the number of consecutive seconds that the upstream or downstream data rate needs to remain above the Upstream or Downstream Exit Bitrate Threshold in order to determine that a transition out of Energy Management DLS Mode is required.

G.2.6.4 CmMac

This object contains attributes of each CM MAC interface.

References: [RFC 4546] docsIfCmMacTable

Table 182 - CmMac Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
ifIndex	InterfaceIndex	Key	InterfaceIndex of MAC Domain interface		
CmtsAddr	MacAddress	R/O			
Capabilities	EnumBits	R/O	atmCells(0) concatenation(1)		false
RangingRespTimeout	TimeTicks	R/W			20
RangingTimeout	TimeInterval	R/W		HundredOfSeconds	20

G.2.6.4.1 ifIndex

This key represents the interface index of the MAC Domain to which this instance applies.

G.2.6.4.2 CmtsAddr

This attribute 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.

G.2.6.4.3 Capabilities

This attribute identifies the capabilities of the MAC implementation at this interface. Note that packet transmission is always supported. Therefore, there is no specific bit required to explicitly indicate this capability.

G.2.6.4.4 RangingRespTimeout

This attribute identifies the waiting time for a Ranging Response packet. This attribute has been obsoleted and replaced by RangingTimeout to correct the typing to TimeInterval.

G.2.6.4.5 RangingTimeout

This attribute identifies the waiting time for a Ranging Response packet. This attribute replaces the obsoleted RangingRespTimeout attribute.

G.2.6.5 RbaRptCfg

This object contains FDX RBA reporting configuration attributes of each FDX enable interface.

Table 183 - RbaRptCfg Object Attributes

Attribute Name	Type	Access	Type Constraints	Units	Default
Enable	Boolean	R/W	Enable/Disable RBA reporting		false
SchedStartTime	TimeStamp	R/W			
Duration	UnsignedByte	R/W	1..255	seconds	30

G.2.6.5.1 Enable

This attribute immediately begins RBA statistics collection. The collection duration default is 30 seconds.

G.2.6.5.2 StartTime

This attribute schedules RBA statistics collection.

G.2.6.5.3 Duration

This attribute specifies the RBA statistics collection period duration.

Appendix I Spectrum Analysis Use Cases (Informative)

This appendix describes several use cases where the Signal Quality Monitoring features introduced in DOCSIS 3.0 can be utilized to manage the HFC plant.

To maintain the HFC network in optimal conditions constant monitoring of the physical characteristics is desired. This practice helps in the early detection of plant problems. These problems, if not properly corrected could cause degradation of services that are offered over the DOCSIS network. The RF impairments may often be the root cause of the problem affecting the quality of services offered over DOCSIS. These impairments result in excessive logging, and poor statistics indicating a lower quality of experience for customer of the services.

Ideally, rather than inferring the presence of RF impairments in the HFC from DOCSIS MAC statistics (for example), the use of Signaling Quality measurement equipment dedicated to monitor the HFC spectrum is desired. However, the cost of such equipment and its associated management and operation may not be justifiable. Instead, active network elements such as CMTSs have evolved their capabilities to report RF measurements using an SNMP management interface. The main advantage of this approach is the constant availability of information across the network. Such information can be correlated to determine e.g., a group of CMs with a common tap in the HFC path reporting the same measurements problem. The signal monitoring approach is similar to how specialized equipment is used to further isolate the problems based on the coarse measurements from a CMTS.

This appendix describes use cases for two main categories of the Enhanced Signaling Quality Monitoring features of DOCSIS 3.0:

- Normalization of RF Impairment Measurements
- Spectrum Amplitude Measurements for Upstream Interfaces

I.1 Normalization of RF Impairment Measurements

DOCSIS [RFC 4546] provides SNR (Signal-to-Noise) measurement. SNR among other measurements are available on a per CM basis and per interface.

SNR values reported may not be uniform amongst different CMTS vendors. Therefore, it might not be possible to compare and analyze information from different devices to determine the HFC plant conditions.

Major contributors to impairments in the DOCSIS channels are linear distortion, non-linear distortion, impulse noise, and ingress noise.

DOCSIS pre-equalization provides a mechanism to correct the linear distortion of each individual CM transmission. Ingress noise robustness has no specification requirements beyond the assumed RF plant conditions in [PHYv4.0]. However, vendors have provided mechanisms to mitigate noise and ingress interference in plants that have more severe noise conditions than the ones assumed in the [PHYv4.0] specification.

The available RF measurements in DOCSIS 3.0 are listed in Table 184 where the DOCSIS 3.0 added features are indicated in **bold** text and are the basis for the use cases of this section. In general, downstream RF measurements are performed by individual CMs while the upstream measurements are performed by the CMTS either at an interface or at a CM level. Based on CMTS and CM interactions, the CM provides an indirect measure of the distortion in the upstream channel through its pre-equalization coefficients.

Table 184 - RF Management Statistics Available

CM (Downstream Measurements)	CMTS (Upstream Measurements)	Measurements Categories
SNR	SNR	Noise conditions
RxMER	**RxMER**	
	CNIR	
	Expected Received Power	Power level
Correctable/uncorrectable errors	Correctable/uncorrectable errors per CM	FEC performance statistics
	Correctable/uncorrectable errors per US interface	

CM (Downstream Measurements)	CMTS (Upstream Measurements)	Measurements Categories
Downstream micro-reflections	Upstream micro-reflections per CM	Linear distortion
CM post-equalization data	CM pre-equalization[1]	
Note: [1]CM may provide more accurate pre-equalization coefficient than what the CMTS is able to calculate.		

The following use cases refer to the noise measurement enhancements for DOCSIS 3.0.

I.1.1 Use Case 1: Figure of Merit Estimation for Logical Upstream Channel

This Use Case defines a Figure of Merit for Logical Upstream Channel measurement that an operator can use to periodically collect information to characterize the performance of the HFC part of the Cable distribution network.

To overcome non-uniform SNR measurements, DOCSIS 3.0 defines two measurements: RxMER (Receive Modulation Error Rate) and CNIR (Carrier to Noise plus Interference Ratio). These provide better indication of the HFC plant impairments and the corrections achieved by the CMTS through compensation techniques. Combining RxMER and CNIR, a Figure of Merit of impairment compensation efficiency can be defined when noise or interference is present.

RxMER measures the average quantization error just prior to FEC, and CNIR measures the carrier to noise plus interference ratio prior to demodulation. A Figure of Merit of how efficiently interference and distortion is compensated in a logical channel can be defined as:

Figure of Merit (logical channel) = RxMER - CNIR

The variables from Annex D to retrieve are:

- RxMER: docsIf3SignalQualityExtRxMER
- CNIR: docsIf3CmtsSignalQualityExtCNIR

The Figure of Merit is relevant when the device is capable of suppressing ingressors, thus increasing the RxMER value with respect to the channel CNIR.

To minimize the uncertainties in measuring the Figure of Merit due to distortion that is unique to individual upstream paths between a CM and CMTS, it is advisable to operate with pre-equalization on (see docsIfUpChannelPreEqEnable of [RFC 4546]).

I.1.2 Use Case 2: Figure of Merit Estimation per CM

This Use Case defines a Figure of Merit per CM transmission. Similar to Use Case 1, the operator can periodically collect information to characterize the performance of CMs in terms of figure of Merit for the given CMTS the CM is attached to.

Unlike RxMER, the SNR parameter is unique for each CM. This allows you to define a Figure of Merit on a per CM basis. A Figure of Merit of how efficiently interference and distortion affecting a CM is compensated can be defined as:

Figure of Merit (CM) = SNR (CM) - CNIR (of the logical upstream channel)

The variables from [DOCS-IF3-MIB] and Annex D to retrieve are:

- SNR: docsIf3CmtsCmUsStatusSignalNoise
- CNIR: docsIf3CmtsSignalQualityExtCNIR

This Figure of Merit indicates if a CM, through its pre-equalization mechanism, is efficiently compensating the linear distortion in its upstream path.

I.1.3 Use Case 3: Absolute Noise and Interference Estimation

Traditionally CMTSs are expected to command the CMs' power transmission so that the CMTS received power is close to 0 dBmV across all CMs.

This Use Case defines how an operator may derive the absolute value of the noise plus interference (in dBmV) from the reported value (CNIR in dB) which is a relative measure.

For example, CNIR and ExpectedRxSignalPower can be used to estimate noise and interference levels (N+I) across the operator's network in dBmV as:

N + I = CNIR - ExpectedRxSignalPower (CMs of the logical upstream channel)

Operators may determine the difference between the target and the actual received power at the CMTS using the following equation:

CM Offset Power = CM Rx Power - ExpectedRxSignalPower

The variables from [DOCS-IF3-MIB] and Annex D to retrieve are:

- CM Rx Power: docsIf3CmtsCmUsStatusRxPower
- ExpectedRxSignalPower: docsIf3CmtsSignalQualityExtExpectedRxSignalPower

Operators may estimate individual CM CNIR by combining the CNIR obtained for the logical channel and the CM offset power as follows:

CM Estimated CNIR = CM Offset Power + CNIR

CM Offset Power: The difference between the actual received CM power level and the expected commanded received signal power at the CMTS.

The variables from [DOCS-IF3-MIB] and Annex D to retrieve are:

- CNIR: docsIf3CmtsSignalQualityExtCNIR
- CM Rx Power: docsIf3CmtsCmUsStatusRxPower
- Expected Commanded Received Signal Power: docsIf3CmtsSignalQualityExtExpectedRxSignalPower

Appendix II Acknowledgements (Informative)

On behalf of the cable industry and our member companies, CableLabs would like to thank the following individuals for their contributions to the development of this specification.

Contributor	Company Affiliation
Mark Lynch	Arris
Dan Torbet	Arris, Commscope
Dwain Frieh	Commscope
Bruce Currivan	Broadcom
Roger Fish	Broadcom
Thomas Clack	Broadcom
Niem Dang	Time Warner Cable
Kirk Erichsen	Time Warner Cable, OAM Technology Consulting
Steve Burroughs	CableLabs
Miguel Alvarez	CableLabs
Brian Hedstrom	OAM Technology Consulting
Kevin Luehrs	CableLabs, OAM Technology Consulting
Pawel Sowinski	Cisco, Falcon V Systems
Dan Hegglin	Cisco
Joe Solomon	Comcast
John Bevilacqua	Comcast
Larry Wolcott	Comcast
Andrew Sundelin	Dial in the Sun, LLC
Tom Staniec	GainSpeed
Hesham ElBakoury	Huawei
Satish Mudugere	Intel, Maxlinear
Mukul Joshi	ST Micro

Appendix III Revision History (Informative)

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I02-200311.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-20.2070-6	2/27/2020	DOCSIS 4.0 CM OSSI I02 Compilation	Burroughs

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I03-210127.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-20.2141-3	12/17/2020	DOCSIS 4.0 CM OSSI I03 Compilation	Burroughs

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I04-210521.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-21.2163-4	5/13/2021	DOCSIS 4.0 CM OSSI I04 compilation	Burroughs

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I05-210927.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-21.2192-1	9/16/2021	DOCSIS 4.0 CM OSSI I05 compilation	Burroughs

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I06-220302.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-22.2227-1	3/2/2022	DOCSIS 4.0 CM OSSI I06 compilation	Burroughs

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I07-221116.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-22.2285-1	10/20/2022	DOCSIS 4.0 CM OSSI I07 compilation	Burroughs

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I08-230516.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-23.2303-2	4/20/2023	Compilation for CM-OSSIv4.0 I08	Erichsen

The following Engineering Change was incorporated into CM-SP-CM-OSSIv4.0-I09-231012.

ECN Identifier	Accepted Date	Title of EC	Author
CM-OSSIv4.0-N-23.2324-3	9/7/2023	CM-OSSIv4.0 I09	Erichsen

* * *