Data-Over-Cable Service Interface Specifications
DCA - MHAv2

Remote PHY Specification

CM-SP-R-PHY-I06-170111

ISSUED

Notice

This DOCSIS® specification is the result of a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc. for the benefit of the cable industry and its customers. You may download, copy, distribute, and reference the documents herein only for the purpose of developing products or services in accordance with such documents, and educational use. Except as granted by CableLabs® in a separate written license agreement, no license is granted to modify the documents herein (except via the Engineering Change process), or to use, copy, modify or distribute the documents for any other purpose.

This document may contain references to other documents not owned or controlled by CableLabs. Use and understanding of this document may require access to such other documents. Designing, manufacturing, distributing, using, selling, or servicing products, or providing services, based on this document may require intellectual property licenses from third parties for technology referenced in this document. To the extent this document contains or refers to documents of third parties, you agree to abide by the terms of any licenses associated with such third-party documents, including open source licenses, if any.

© Cable Television Laboratories, Inc., 2014-2017

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-R-PHY-I06-170111
Document Title	Remote PHY Specification
Revision History	I01 - Released 06/15/2015 I02 - Released 10/01/2015 I03 - Released 01/21/2016 I04 - Released 05/12/2016 I05 - Released 09/23/2016 I06 - Released 01/11/2017
Date:	January 11, 2017
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, and may include several alternative solutions for consideration.
Draft	A document in Specification format considered largely complete, but lacking review by Members and Technology Suppliers. Drafts are susceptible to substantial change during the review process.
Issued	A generally public document that has undergone Member and Technology Supplier review, cross-vendor interoperability, and is available for Certification testing. Issued Specifications are subject to the Engineering Change (EC) Process.
Closed	A static document, reviewed, tested, validated, and closed to further ECs.

Trademarks

CableLabs® is a registered trademark of Cable Television Laboratories, Inc. Other CableLabs marks are listed at http://www.cablelabs.com/certqual/trademarks. All other marks are the property of their respective owners.

Table of Contents

Figures

Tables

This page intentionally left blank

1 SCOPE

1.1 Introduction and Purpose

Modular Headend Architecture version 2 (MHAv2)/Remote PHY technology allows a CMTS to support an IP-based digital HFC plant. In an IP-based digital HFC plant, the fiber portion utilizes a baseband network transmission technology such as Ethernet, EPON (Ethernet over Passive Optical Networks), GPON (Gigabit Passive Optical Network), or any Layer 2 technology that would support a fiber-based Layer 1. MHAv2 uses a Layer 3 pseudowire between a CCAP Core and a series of Remote PHY devices. One of the common locations for a Remote PHY device at an optical node device that is located at the junction of the fiber and coax plants.

1.2 MHAv2 Interface Documents

A list of the documents in the MHAv2 family of specifications is provided below. For updates, refer to http://www.cablelabs.com/specs/specification-search/.

Table 1 - List of MHAv2 Specifications

Designation	Title
CM-SP-R-PHY	Remote PHY Specification
CM-SP-R-DEPI	Remote Downstream External PHY Interface Specification
CM-SP-R-UEPI	Remote Upstream External PHY Interface Specification
CM-SP-GCP	Generic Control Plane Specification
CM-SP-R-DTI	Remote DOCSIS Timing Interface Specification
CM-SP-R-OOB	Remote Out-of-Band Specification
CM-SP-R-OSSI	Remote PHY OSS Interface Specification

NOTE: MHAv2 does not explicitly use any of the original Modular Headend Architecture specifications.

1.3 Requirements and 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 to read and the LSB being the last bit to read.

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.

2 REFERENCES

At the time of publication, the editions indicated were valid. All references are subject to revision, and users of this document are encouraged to investigate the possibility of applying the most recent editions of the documents listed below. References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific. For a nonspecific reference, the latest version applies.

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-I15-170111, January 11, 2017, Cable Television Laboratories, Inc.
[CCAP-OSSIv3.1]	DOCSIS 3.1 CCAP OSSI Specification, CM-SP-CCAP-OSSIv3.1-I08-170111, January 11, 2017, Cable Television Laboratories, Inc.
[CM-OSSIv3.1]	DOCSIS 3.1 Cable Modem OSSI Specification, CM-SP-CM-OSSIv3.1-I08-170111, January 11, 2017, Cable Television Laboratories, Inc.
[DEPI]	Downstream External PHY Interface Specification, CM-SP-DEPI-I08-100611, June 11, 2010, Cable Television Laboratories, Inc.
[DRFI]	DOCSIS Downstream Radio Frequency Interface, CM-SP-DRFI-I16-170111, January 11, 2017, Cable Television Laboratories, Inc.
[FIPS 140-2]	Federal Information Processing Standards Publication (FIPS PUB) 140-2, Security Requirements for Cryptographic Modules, May 2001.
[FIPS 180-4]	Federal Information Processing Standards Publication (FIPS PUB) 180-4, Secure Hash Standard, May 2014.
[GCP]	Generic Control Plane Specification, CM-SP-GCP-I02-160512, May 12, 2016, Cable Television Laboratories, Inc.
[IANA-PORTS]	IANA, Port Numbers, June 2004.
[IEEE 802.1ae]	IEEE Std 802.1ae-2006, IEEE Standard for Local and metropolitan area networks-Media Access Control (MAC) Security, August 2006.
[IEEE 802.1q]	IEEE Std 802.1q-2003, Virtual Bridged Local Area Networks, May 2003.
[IEEE 802.1x]	IEEE Std 802.1x-2010, IEEE Standard for Local and metropolitan area networks--Port-Based Network Access Control, February 2010.
[IEEE 802.3]	IEEE Std 802.3™-2002, Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, March 2002.
[IEEE 1588]	IEEE Std 1588-2008, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, July 2008.
[ISO 13818-1]	ISO/IEC 13818-1:2013, Information Technology - Generic Coding of Moving Pictures and Associated Audio Information. Part 1: System, May 23, 2013.
[ISO/IEC-61169-24]	ISO/IEC-61169-24, Radio-frequency connectors - Part 24: Sectional specification - Radio frequency coaxial connectors with screw coupling, typically for use in 75 ohm cable distribution systems (type F), 2001.
[ITU-T J.83]	ITU-T Recommendation J.83 (4/97), Digital multi-programme systems for television sound and data services for cable distribution.
[MULPIv3.0]	DOCSIS MAC and Upper Layer Protocols Interface Specification, CM-SP-MULPIv3.0-I30-170111, January 11, 2017, Cable Television Laboratories, Inc.

[MULPIv3.1]	DOCSIS MAC and Upper Layer Protocols Interface Specification, CM-SP-MULPIv3.1-I10-170111, January 11, 2017, Cable Television Laboratories, Inc.
[PHYv3.0]	DOCSIS 3.0 Physical Layer Specification, CM-SP-PHYv3.0-I13-170111, January 11, 2017, Cable Television Laboratories, Inc.
[PHYv3.1]	DOCSIS 3.1 Physical Layer Specification, CM-SP-PHYv3.1-I10-170111, January 11, 2017, Cable Television Laboratories, Inc.
[SECv3.0]	DOCSIS 3.0 Security Specification, CM-SP-SECv3.0-I16-160602, June 2, 2016, Cable Television Laboratories, Inc.
[SECv3.1]	DOCSIS 3.1 Security Specification, CM-SP-SECv3.1-I07-170111, January 11, 2017, Cable Television Laboratories, Inc.
[PKCS#7]	RSA Laboratories, PKCS #7: Cryptographic Message Syntax Standard, An RSA Laboratories Technical Note, Version 1.5, Revised November 1, 1993.
[R-DEPI]	Remote Downstream External PHY Interface Specification, CM-SP-R-DEPI-I06-170111, January 11, 2017, Cable Television Laboratories, Inc.
[R-DTI]	Remote DOCSIS Timing Interface Specification, CM-SP-R-DTI-I04-170111, January 11, 2017, Cable Television Laboratories, Inc.
[R-OOB]	Remote Out-of-Band Specification, CM-SP-R-OOB-I05-170111, January 11, 2017, Cable Television Laboratories, Inc.
[R-OSSI]	Remote PHY OSS Interface Specification, CM-SP-R-OSSI-I054-170111, January 11, 2017, Cable Television Laboratories, Inc.
[R-UEPI]	Remote Upstream External PHY Interface Specification, CM-SP-R-UEPI-I05-170111, January 11, 2017, Cable Television Laboratories, Inc.
[RFC 768]	IETF RFC 768, User Datagram Protocol, August 1980.
[RFC 791]	IETF RFC 791, Internet Protocol-DARPA, September 1981.
[RFC 868]	IETF RFC 768, Time Protocol, May 1983.
[RFC 1191]	IETF RFC 1191, MTU Path Discovery, November 1990.
[RFC 1350]	IETF RFC 1350, The TFTP Protocol (Revision 2), July 1992.
[RFC 1700]	IETF RFC 1700, Assigned Numbers, October 1994.
[RFC 1945]	IETF RFC 1945, Hypertext Transfer Protocol -- HTTP/1.0, May 1996.
[RFC 1981]	IETF RFC 1981, Path MTU Discovery for IP version 6, August 1996.
[RFC 2131]	IETF RFC 2131, Dynamic Host Configuration Protocol, March 1997.
[RFC 2348]	IETF RFC 2348, TFTP Blocksize Option, May 1998.
[RFC 2597]	IETF RFC 2597, Assured Forwarding PHB Group, June 1999.
[RFC 2616]	IETF RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1, June 1999.
[RFC 2863]	IETF RFC 2863, The Interfaces Group MIB, K. McCloghrie, F. Kastenholz, June 2000.
[RFC 2983]	IETF RFC 2983, Differentiated Services and Tunnels, October 2000.
[RFC 3246]	IETF RFC 3246, An Expedited Forwarding PHB (Per-Hop Behavior), March 2002.
[RFC 3260]	IETF RFC 3260, New Terminology and Clarifications for Diffserv, April 2002.
[RFC 3308]	IETF RFC 3308, Layer Two Tunneling Protocol (L2TP) Differentiated Services Extension, November 2002.
[RFC 3748]	IETF RFC 3748, Extensible Authentication Protocol (EAP)
[RFC 3931]	IETF RFC 3931, Layer Two Tunneling Protocol - Version 3 (L2TPv3), March 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 4293]	IETF RFC 4293, Management Information Base for the Internet Protocol (IP), S. Routhier, April 2006.
[RFC 4307]	IETF RFC 4307, Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2), December 2005.
[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 5216]	IETF RFC 5216, IEAP-TLS Authentication Protocol, March 2008.
[RFC 5247]	IETF RFC 5247, EAP Key Management Framework, August 2008.
[RFC 5280]	IETF RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008.
[RFC 5601]	IETF RFC 5601, Pseudowire (PW) Management Information Base (MIB), T. Nadeau, D. Zelig, July 2009.
[RFC 6960]	IETF RFC 6960, I.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP, June 2013.
[RFC 7296]	IETF RFC 7296, Internet Key Exchange Protocol Version 2 (IKEv2), October 2014.
[RSA 1]	RSA Laboratories, PKCS #1: RSA Encryption Standard. Version 1.5, RSA Security, Inc., Bedford, MA, November 1993.
[RSA 3]	RSA Laboratories, PKCS #3: Diffie-Hellman Key Agreement Standard, Version 1.4, RSA Security, Inc., Bedford, MA, November 1993.
[SCTE 02]	ANSI/SCTE 02, Specification for "F" Port, Female Indoor, 2006.
[Vendor ID]	Refers to RFC 3232 “Assigned Number” by the IETF, Jan 2002. This spec refers to the IANA web page which is http://www.iana.org/assignments/enterprise-numbers.
[X.509]	ITU-T Recommendation X.509: Information Technology - Open Systems Interconnection - The Directory: Public-key and Attribute Certificate Frameworks, March 2000.

2.2 Informative References

This document uses the following informative references:

[IANA-L2TP]	IANA, Layer Two Tunneling Protocol (L2TP) Parameters.
[ISO 8802-2]	ISO/IEC 8802-2: 1994 (IEEE Std 802.2: 1994) - Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 2: Logical link control.
[RFC 3140]	IETF RFC 3140, Per Hop Behavior Identification Codes, June 2001.

2.3 Reference Acquisition

- 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
- Federal Information Processing Standards: 100 Bureau Drive, Mail Stop 3200, Gaithersburg, MD 20899-3200. Phone +1-301-975-4054; Fax +1-301-926-8091. http://csrc.nist.gov/publications/fips/.
- The Institute of Electrical and Electronics Engineers, Inc., Internet: http://standards.ieee.org
- International Organization for Standardization (ISO), Tel.: +41 22 749 02 22, Fax: +41 22 749 01 55, www.standardsinfo.net
- Internet Assigned Numbers Authority, IANA, Internet: http://www.iana.org
- 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.

- ITU Recommendations: Place des Nations, CH-1211, Geneva 20, Switzerland. Phone +41-22-730-51-11; Fax +41-22-733-7256. http://www.itu.int.
- Public Key Cryptography Standards: RSA Security Inc. 174 Middlesex Turnpike, Bedford, MA 01730. Phone +1-781-515-5000; Fax 781-515-5010. http://www.rsasecurity.com/rsalabs/.
- SCTE, Society of Cable Telecommunications Engineers, 140 Philips Road, Exton, PA 19341-1318, Phone+1-800-542-5040; Fax+1-610-363-5898. http://www.scte.org/default.aspx/.

3 TERMS AND DEFINITIONS

This specification uses the following terms:

Bonded Channels	A logical channel comprising multiple individual channels.
Cable Modem	A modulator-demodulator at subscriber locations intended for use in conveying data communications on a cable television system.
CCAP Core	A CCAP device that uses MHAv2 protocols to interconnect to an RPD.
Converged Interconnect Network	The network (generally gigabit Ethernet) that connects a CCAP Core to an RPD.
Customer Premises Equipment	Equipment at the end user's premises; may be provided by the service provider.
Data Rate	Throughput, data transmitted in units of time usually in bits per second (bps).
Decibels	Ratio of two power levels expressed mathematically as $dB = 10\log_{10}(P_{OUT}/P_{IN})$.
Decibel-Millivolt	Unit of RF power expressed in decibels relative to 1 millivolt, where $dBmV = 20\log^{10}$(value in mV/1 mV).
Downstream	• Transmissions from CMTS to CM. This includes transmission from the CCAP Core to the EQAM, as well as the RF transmissions from the EQAM to the CM • RF spectrum used to transmit signals from a cable operator's headend or hub site to subscriber locations.
Dynamic Host Configuration Protocol	A network protocol enabling a server to automatically assign an IP address to a network element.
Edge QAM Modulator	A headend or hub device that receives packets of digital video or data. It re-packetizes the video or data into an MPEG transport stream and digitally modulates the digital transport stream onto a downstream RF carrier using quadrature amplitude modulation (QAM).
Flow	A stream of packets in DEPI used to transport data of a certain priority from the CCAP Core to a particular QAM channel of the EQAM. In PSP operation, there can exist several flows per QAM channel.
Gbps	Gigabits per second
Gigahertz	A unit of frequency; 1,000,000,000 or 10^9 Hz
GigE	Gigabit Ethernet (1 Gbps)
Hertz	A unit of frequency; formerly cycles per second
Hybrid Fiber/Coax System	A broadband bidirectional shared-media transmission system using optical fiber trunks between the headend and the fiber nodes, and coaxial cable distribution from the fiber nodes to the customer locations.
Institute of Electrical and Electronic Engineers	A voluntary organization which, among other things, sponsors standards committees and is accredited by the American National Standards Institute (ANSI).
Internet Engineering Task Force	A body responsible for, among other things, developing standards used in the Internet.
Internet Protocol	An Internet network-layer protocol
kilohertz	Unit of frequency; 1,000 or 10^3 Hz; formerly kilocycles per second
L2SS	Layer 2 Specific Sublayer. DEPI is an L2SS of L2TPv3.

L2TP Access Concentrator	If an L2TP Control Connection Endpoint (LCCE) is being used to cross-connect an L2TP session directly to a data link, we refer to it as an L2TP Access Concentrator (LAC). An LCCE may act as both an L2TP Network Server (LNS) for some sessions and an LAC for others, so these terms must only be used within the context of a given set of sessions unless the LCCE is, in fact, single purpose for a given topology.
L2TP Attribute Value Pair	The L2TP variable-length concatenation of a unique Attribute (represented by an integer), a length field, and a Value containing the actual value identified by the attribute.
L2TP Control Connection	An L2TP control connection is a reliable control channel that is used to establish, maintain, and release individual L2TP sessions, as well as the control connection itself.
L2TP Control Connection Endpoint	An L2TP node that exists at either end of an L2TP control connection. May also be referred to as an LAC or LNS, depending on whether tunneled frames are processed at the data link (LAC) or network layer (LNS).
L2TP Control Connection ID	The Control Connection ID field contains the identifier for the control connection, a 32-bit value. The Assigned Control Connection ID AVP, Attribute Type 61, contains the ID being assigned to this control connection by the sender. The Control Connection ID specified in the AVP must be included in the Control Connection ID field of all control packets sent to the peer for the lifetime of the control connection. Because a Control Connection ID value of 0 is used in this special manner, the zero value must not be sent as an Assigned Control Connection ID value.
L2TP Control Message	An L2TP message used by the control connection.
L2TP Data Message	Message used by the data channel.
L2TP Endpoint	A node that acts as one side of an L2TP tunnel.
L2TP Network Server	If a given L2TP session is terminated at the L2TP node and the encapsulated network layer (L3) packet processed on a virtual interface, we refer to this L2TP node as an L2TP Network Server (LNS). A given LCCE may act as both an LNS for some sessions and an LAC for others, so these terms must only be used within the context of a given set of sessions unless the LCCE is in fact single purpose for a given topology.
L2TP Pseudowire	An emulated circuit as it traverses a packet-switched network. There is one Pseudowire per L2TP Session.
L2TP Pseudowire Type	The payload type being carried within an L2TP session. Examples include PPP, Ethernet, and Frame Relay.
L2TP Session	An L2TP session is the entity that is created between two LCCEs in order to exchange parameters for and maintain an emulated L2 connection. Multiple sessions may be associated with a single Control Connection.
L2TP Session ID	A 32-bit field containing a non-zero identifier for a session. L2TP sessions are named by identifiers that have local significance only. That is, the same logical session will be given different Session IDs by each end of the control connection for the life of the session. When the L2TP control connection is used for session establishment, session IDs are selected and exchanged as Local Session ID AVPs during the creation of a session. The Session ID alone provides the necessary context for all further packet processing, including the presence, size, and value of the Cookie, the type of L2-Specific Sublayer, and the type of payload being tunneled.
MAC Domain	A grouping of Layer 2 devices that can communicate with each other without using bridging or routing. In DOCSIS, it is the group of CMs that are using upstream and downstream channels linked together through a MAC forwarding entity.
Maximum Transmission Unit	Maximum size of the Layer 3 payload of a Layer 2 frame.
Mbps	Megabits per second

Media Access Control	Used to refer to the Layer 2 element of the system which would include DOCSIS framing and signaling.
Megahertz	A unit of frequency; 1,000,000 or 10^6 Hz
Modulation Error Ratio	The ratio of average signal constellation power to average constellation error power - that is, digital complex baseband signal-to-noise ratio - expressed in decibels.
Microsecond	10^{-6} second
Millisecond	10^{-3} second
Modulation Error Ratio	The ratio of the average symbol power to average error power.
Multiple System Operator	A corporate entity that owns and/or operates more than one cable system.
Nanosecond	10^{-9} second
Packet Identifier	PID (system): A unique integer value used to identify elementary streams of a program in a single or multi-program Transport Stream as described in section 2.4.3 of ITU-T Rec. H.222.0 [ISO 13818-1]
Physical Media Dependent Sublayer	A sublayer of the Physical layer which is concerned with transmitting bits or groups of bits over particular types of transmission link between open systems and which entails electrical, mechanical, and handshaking procedures.
Pilot tones	Required in the HFC network to ensure that amplifiers in the network are operating correctly. Amplifiers use these tones to adjust gain and keep signals at the appropriate output level.
Precision Time Protocol	A protocol used to synchronize clocks throughout a network.
Program Clock Reference	A timestamp in the Video Transport Stream from which decoder timing is derived.
Pseudowire	An IP tunnel between two points in an IP network.
QAM channel	Analog RF channel that uses quadrature amplitude modulation (QAM) to convey information
Quadrature Amplitude Modulation	A modulation technique in which an analog signal's amplitude and phase vary to convey information, such as digital data.
Radio Frequency	In cable television systems, this refers to electromagnetic signals in the range 5-1000 MHz.
Radio Frequency Interface	Term encompassing the downstream and the upstream radio frequency interfaces.
Request For Comments	A technical policy document of the IETF; these documents can be accessed on the World Wide Web at http://www.rfc-editor.org/.
Request-Grant Delay Time	The time from when a CM requests bandwidth, using an uncontended bandwidth request (REQ), to when it receives a MAP message with the granted transmit opportunity in it.
Remote-PHY Device	The Remote-PHY Device (RPD) is a device in the network which implements the Remote-PHY specification to provide conversion from digital Ethernet transport to analog RF transport.
Session	An L2TP data plane connection from the CCAP Core to the QAM channel. There must be one session per QAM Channel. There is one DEPI pseudowire type per session. There may be one MPT flow or one or more PSP flows per session. Multiple sessions may be bound to a single control connection.
StopCCN	L2TPv3 Stop-Control-Connection-Notification message
Trivial File Transfer Protocol	A file transfer protocol. Generally used for automated transfer of configuration or boot files between machines

Upconverter	A device used to change the frequency range of an analog signal, usually converting from a local oscillator frequency to an RF transmission frequency.
Upstream	• Transmissions from CM to CMTS. This includes transmission from the EQAM to CCAP Core as well as the RF transmissions from the CM to the EQAM. • RF spectrum used to transmit signals from a subscriber location to a cable operator's headend or hub site.
Upstream Channel Descriptor	The MAC Management Message used to communicate the characteristics of the upstream physical layer to the cable modems.
Video on Demand System	System that enables individuals to select and watch video content over a network through an interactive television system.

4 ABBREVIATIONS AND ACRONYMS

This specification uses the following abbreviations:[1]

ACK	L2TPv3 Explicit Acknowledgement message
ADC	Analog-to-Digital Converter
AF	Assured Forwarding
AGC	Automatic Gain Control
API	Application Programming Interface
ATM	Asynchronous Transfer Mode
AVP	L2TPv3 Attribute Value Pair
BPI	Baseline Privacy Interface
CA	Certificate Authority
CAK	Connectivity Association Key
CCAP	Converged Cable Access Platform
CDN	L2TPv3 Call-Disconnect-Notify message
CIN	Converged Interconnect Network
CLI	Command Line Interface
CM	Cable Modem
CMCI	Cable Modem to Customer Premises Equipment Interface
CMTS	Cable Modem Termination System
CPE	Customer Premises Equipment
CRC	Cyclic Redundancy Check
CRC16	CRC of length 16
CRL	Certificate Revocation List
CSMA	Carrier Sense Multiple Access
CVC	Code Verification Certificate
CVS	Code Verification Signature
CW	Continuous Wave
DAC	Digital-to-Analog Converter
dB	Decibels
dBmV	Decibel-Millivolt
DCA	Distributed CCAP Architecture
DEPI	Downstream External PHY Interface
DER	Distinguished Encoding Rules
DF	Don’t Fragment (bit)
DHCP	Dynamic Host Configuration Protocol
DHCPv4	Dynamic Host Configuration Protocol version 4
DHCPv6	Dynamic Host Configuration Protocol version 6
DOCSIS	Data-Over-Cable Service Interface Specifications

[1] Revised per R-PHY-N-16.1570-1 on 8/22/16 by JB.

DOCSIS-MPT (D-MPT)	DOCSIS MPT Mode
DPI	SCTE-35/Digital Program Insertion
DRFI	Downstream Radio Frequency Interface
DS	Downstream
DSA	Dynamic Service Flow Add
DSCP	Differentiated Services Code Point
DSC	Dynamic Service Flow Change
DSD	Dynamic Service Flow Delete
DTA	Digital Television Adapter
DTI	DOCSIS Timing Interface
DTS	DOCSIS Timestamp, 32-bit
EAP	Extensible Authentication Protocol
EAPoL	Extensible Authentication Protocol over LAN
EBIF	Enhanced TV Binary Interchange Format
EF	Expedited Forwarding
EQAM	Edge QAM
ERM	Edge Resource Manager
ERMI	Edge Resource Manager Interface
ETSI	European Telecommunications Standards Institute
FDM	Frequency Division Multiplex
FQDN	Fully Qualified Domain Name
Gbps	Gigabits per second
GCP	Generic Control Plane
GE	Gigabit Ethernet (Gig E)）
GHz	Gigahertz
HDLC	High-Level Data Link Control
HELLO	L2TPv3 Hello message
HFC	Hybrid Fiber/Coax
HMAC	Hash-based Message Authentication Code
Hz	Hertz
I-CCAP	Integrated CCAP
ICCN	L2TPv3 Incoming-Call-Connected message
ICMP	Internet Control Message Protocol
I-CMTS	Integrated CMTS
ICRP	L2TPv3 Incoming-Call-Reply message
ICRQ	L2TPv3 Incoming-Call-Request message
ID	Identifier
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange

IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISO	International Standards Organization
ITU	International Telecommunications Union
ITU-T	Telecommunication Standardization Sector of the International Telecommunication Union
kbps	Kilobits per second
kHz	Kilohertz
L2SS	Layer 2 Specific Sublayer
L2TP	Layer 2 Transport Protocol
L2TPv3	Layer 2 Transport Protocol version 3
L3	Layer 3
LAC	L2TP Access Concentrator
LCCE	L2TP Control Connection Endpoint
LNS	L2TP Network Server
LSB	Least Significant Bit
MAC	Media Access Control
MAP	Upstream Bandwidth Allocation Map (referred to only as MAP)
Mbps	Megabits per second
MCM	Multi-channel MPEG
M-CMTS	Modular Cable Modem Termination System
MER	Modulation Error Ratio
MHA	Modular Headend Architecture
MHz	Megahertz
MIB	Management Information Base
MKA	MACsec Key Agreement (protocol)
M/N	Relationship of integer numbers M,N that represents the ratio of the downstream symbol clock rate to the DOCSIS master clock rate
MPEG	Moving Picture Experts Group
MPEG-TS	Moving Picture Experts Group Transport Stream
MPT	MPEG-TS mode of R-DEPI
MPTS	Multi Program Transport Stream
ms	Millisecond
MSB	Most Significant Bit
MSK	Master Secret Key
MSO	Multiple System Operator
MSB	Most Significant Bit
MTU	Maximum Transmission Unit
NAD	Network Access Device
NDF	Narrowband Digital Forward
NDR	Narrowband Digital Return
ns	Nanosecond

NSI	Network Side Interface
ONU	Optical Network Unit
OCSP	Online Certificate Status Protocol
OSSI	Operations System Support Interface
PAE	Port Access Entity
PAT	Program Association Table
PCR	Program Clock Reference
PHB	Per Hop Behavior
PHB-ID	Per Hop Behavior Identifier
PHS	Payload Header Suppression
PHY	Physical Layer
PID	Packet Identifier
PKI	Public Key Infrastructure
PMD	Physical Media Dependent Sublayer
PMT	Program Map Table
PMTUD	Path MTU Discovery
PNM	Proactive Network Maintenance
PPP	Point-to-Point Protocol
PSI	Program Specific Information
PSIP	Program and System Information Protocol
PSP	Packet Streaming Protocol
PTP	Precision Time Protocol
PW	Pseudowire
QAM	Quadrature Amplitude Modulation
QAM ch	QAM Channel
RCP	R-PHY Control Protocol
RDC	Regional Data Center
R-DEPI	Remote Downstream External PHY Interface
RF	Radio Frequency
RFI	Radio Frequency Interface
RFC	Request For Comments
ROTs	RCP Objects/TLVs
RPD	Remote PHY Device
RSA	Rivest-Shamir-Adleman (cryptosystem)
R-UEPI	Remote Upstream External PHY Interface
SCCRN	L2TPv3 Start-Control-Connection-Connected message
SCCRP	L2TPv3 Start-Control-Connection-Reply message
SCCRQ	L2TPv3 Start-Control-Connection-Request message
S-CDMA	Synchronous Code Division Multiple Access
SGID	Service Group Identifier
SLI	L2TPv3 Set Link Info message

SPTS	Single Program Transport Stream
SSD	Secure Software Download
StopCCN	L2TPv3 Stop-Control-Connection-Notification message
TCP	Transmission control protocol
TFTP	Trivial File Transfer Protocol
TSID	MPEG2 Transport Stream Identifier
UCD	Upstream Channel Descriptor
UDP	User Datagram Protocol
µs	Microsecond
US	Upstream
UTC	Coordinated Universal Time
VoD	Video On Demand
VoIP	Voice over IP

5 TECHNICAL OVERVIEW

5.1 Introduction

In a Remote PHY Architecture, the classic integrated CCAP (I-CCAP) is separated into two distinct components. The first component is the CCAP Core and the second component is the Remote PHY Device (RPD).

The CCAP Core contains both a CMTS Core for DOCSIS and an EQAM Core for Video. The CMTS Core contains the DOCSIS MAC and the upper layer DOCSIS protocols. This includes all signaling functions, downstream and upstream bandwidth scheduling, and DOCSIS framing. The DOCSIS functionality of the CMTS Core is defined by [MULPIv3.0]. The EQAM Core contains all the video processing functions that an EQAM provides today.

The Remote PHY Device is a physical layer converter whose functions are:

- To convert downstream DOCSIS, MPEG video and OOB signals received from a CCAP Core over a digital medium such as Ethernet or PON to analog for transmission over RF or linear optics.
- To convert upstream DOCSIS, and OOB signals received from an analog medium such as RF or linear optics to digital for transmission over Ethernet or PON to a CCAP Core.

The RPD platform contains mainly PHY related circuitry, such as downstream QAM modulators, upstream QAM demodulators, together with pseudowire logic to connect to the CCAP Core.

It provides a subset of the following external interfaces:

CIN Facing:

- One or more 10G or 1G Ethernet or PON ports

Access Network Facing:

- One or more 10G or 1G Ethernet or PON ports
  - Additional RPDs may be daisy-chained through these ports
- One or more RF ports providing connectivity to the access network
  - RF ports may be unidirectional (for use with an external combiner) or bi-directional (internal combiner)
  - RF port output may be RF over coaxial cable or over analog optics

An example reference implementation based on Ethernet is shown in Figure 1.

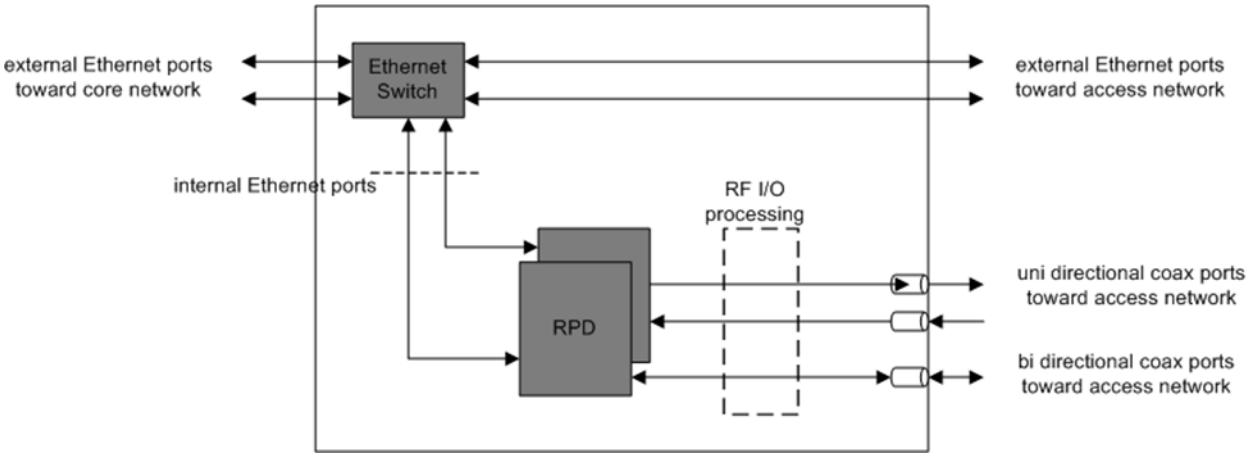

Figure 1 - Logical View of RPD Internals

The DOCSIS functionality of the Remote PHY Device is defined by [PHYv3.1], [MULPIv3.1], [DRFI], and [PHYv3.0].

Together, the CCAP Core and the RPD are the functional equivalent of an I-CCAP (Integrated CCAP), just with different packaging. The MHAv2 specifications describe how the CCAP Core and the RPD interface with each other.

Note that MHAv2 functionality and signaling and DOCSIS functionality and signaling are completely separate. The DOCSIS functionality and signaling remain the same for both I-CMTS and Remote PHY solutions. MHAv2 focuses on a simple deconstruction of the CMTS that moves the CCAP PHY elements into an external RPD device while keeping the DOCSIS CMTS-to-CM signaling untouched.

5.2 System Diagram

Figure 2 shows an abstracted view of a cable operator's network. Note that there are more aggregation points beyond a headend such as a super headend or a regional data center (RDC). For the scope of this specification, the focus will be on the headend aggregation point.

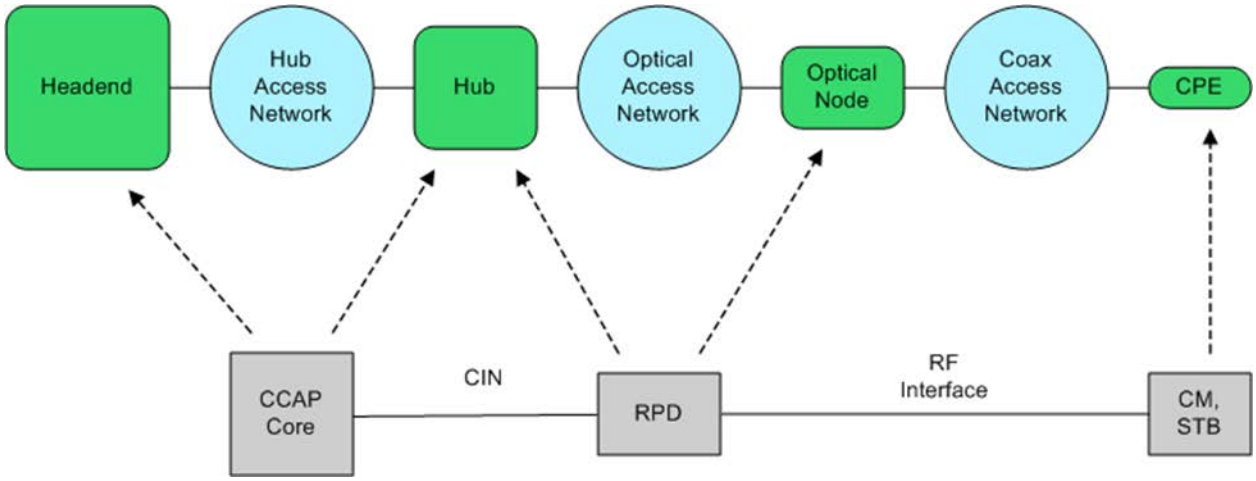

Figure 2 - Remote PHY System Diagram

The items in green are physical locations. The headend is where the majority of the equipment that does not require direct connectivity to the access network resides. Video channel line-ups are often created in the headend. The headend aggregates a number of hubs, with the hub containing equipment that requires direct connectivity to the HFC plant. One example of equipment in the hub is the I-CCAP. The hub aggregates a number of optical nodes. The optical nodes are located in the field and convert between a long point-to-point optical run and a local coax network. The optical node aggregates traffic from a number of subscriber endpoints such as DOCSIS CMs and Video STBs.

5.2.1 Hub Access Network

The hub access network is the network that connects the headend and the hub. The hub access network can be either a switched Layer 2 network or a routed Layer 3 network. It typically is a multi-hop network, which means there can be multiple switches and/or routers between equipment in the headend and the hub.

5.2.2 Optical Access Network

The optical access network is located between the hub and the optical node. The access network has a forward path and a reverse path.

5.2.2.1 Using Linear Optics

The classic HFC plant uses linear optics where the RF spectrum from the coax is modulated onto an optical wavelength. The only type of signal that can traverse this type of network is an RF modulated signal such as a QAM or an OFDM signal.

5.2.2.2 Using Digital Optics Only

A variation of the classic HFC plant uses digital optics in the return path. The RF spectrum is digitized and sampled at the optical node, sent to the headend, and then reconstructed into an analog signal. From the viewpoint of this specification, this will be considered as a subset of a linear optics HFC plant since its operation is transparent to the transmission path which is still a modulated signal such as QAM or OFDMA.

5.2.2.3 Using Digital Optics with IP

A new HFC plant architecture is available that can use any fiber compatible baseband networking technology, such as Ethernet, EPON, or GPON, to drive the fiber portion of the HFC plant. The coax portion of the HFC plant remains the same. With digital optics based upon IP networking, the optical access network could be directly connected from the CCAP Core to the optical node. Since the hub is aggregating many optical nodes, the access network may have one or more network elements in it, where the network elements could be a Layer 2 switch or a Layer 3 router. Note that this model includes network elements that may be physically located at the hub but are connected between the CCAP Core and the optical node.

One of the goals of MHAv2 is to accommodate this new digital IP-based HFC plant architecture while maintaining the minimum impact on the CCAP definition and operation. In this manner, I-CCAP and Remote PHY implementations may be used as needed for different HFC plant architectures while maintaining a common CCAP feature set and software loads.

5.2.3 Coax Access Network

The coax portion of the network is an FDM (frequency division multiplex) plant that carries RF modulated signals. It has an upper frequency bound and a frequency range that is split between the upstream and downstream spectrums.

5.2.4 Location of the Remote PHY Device

For an optical access network based on linear optics, the RPD is located at the hub. For an optical access network based on digital optics, the RPD is located at the optical node.

5.2.5 Location of the CCAP Core

The previous version of I-CCAP is located at the hub where the RF ports can have direct connectivity to the access network. Since the CCAP Core does not have RF ports, this restriction is removed. The CCAP Core can be located at the hub or headend (or at another location beyond the headend, like the regional data center).

The network between the CCAP Core and the RPD is known as the Converged Interconnect Network (CIN). The CIN encompasses either or both the hub access network and the optical access network. The CIN can contain both Layer 2 switches and Layer 3 routers.

5.3 System Architecture

5.3.1 System Components

The reference architecture for a Modular CMTS system is shown in Figure 3. Architectures for video and OOB are similar. This architecture contains both physical and logical components. This section briefly introduces each device and interface.

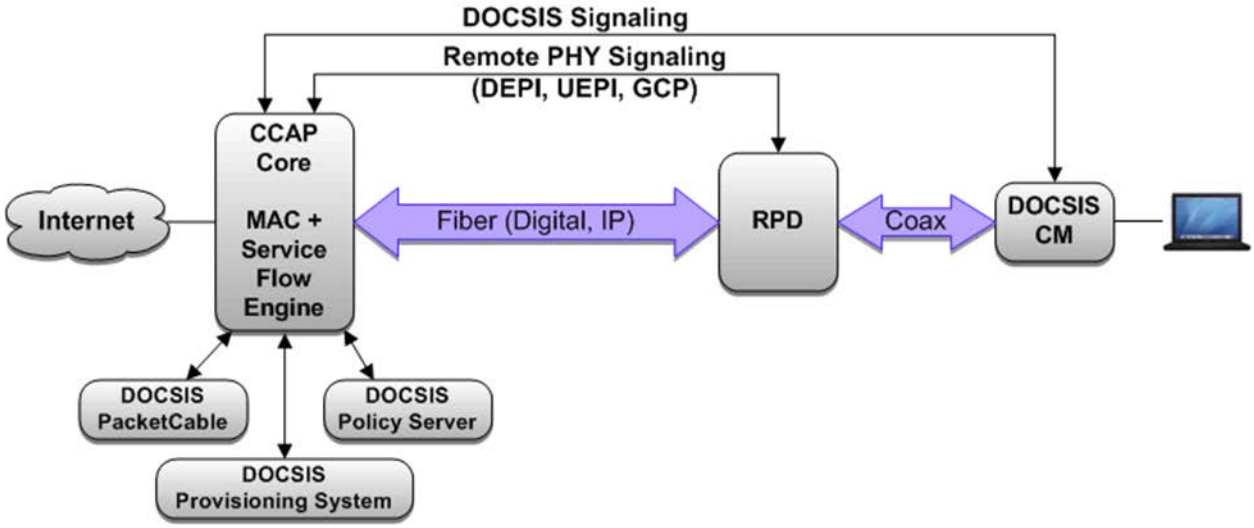

Figure 3 - MHAv2 Reference Architecture for DOCSIS Signaling and Provisioning

The **RPD** is a component that has network interface on one side and an RF interface on the other side. The RPD provides Layer 1 PHY conversion, Layer 2 MAC conversion, and Layer 3 pseudowire support. The RPD RF output may be RF combined with other overlay services such as analog or digital video services.

The **CCAP Core** contains everything a traditional CMTS does, except for functions performed in the RPD. The CCAP Core contains the downstream MAC, the upstream MAC, and all the initialization and operational DOCSIS-related software.

Note that the original MHAv1 architecture had the downstream PHY external and the upstream PHY internal. MHAv1 was used to interface to an EQAM (Edge QAM) device that was co-located at the headend with the CMTS Core. Thus, the main difference between MHAv1 and MHAv2 is the location of the upstream PHY and the role of the solution in the marketplace. From a technical standpoint, the solutions are very similar.

Due to the physical separation of the downstream PHY and the upstream PHY in MHAv1, a DOCSIS Timing Interface (DTI) Server was needed to provide a common frequency of 10.24 MHz and a DOCSIS timestamp between the two MHAv1 elements. In MHAv2, the same DTI server is not required since the downstream and upstream PHYs are co-located in the RPD. A different timing solution referred to as R-DTI is used to provide timing services for functions such as DOCSIS scheduling.

R-DEPI, the Remote Downstream External PHY Interface, is the downstream interface between the CCAP Core and the RPD. More specifically, it is an IP pseudowire between the MAC and PHY in an MHAv2 system that contains both a data path for DOCSIS frames, video packets, and OOB packets, as well as a control path for setting up, maintaining, and tearing down sessions. MHAv1 used the MPT (MPEG-TS) encapsulation. MHAv2 retains the original MPT encapsulation for backward compatibility but also added a new MPEG encapsulation called MCM (Multi-channel MPEG). MHAv2 also requires the PSP (Packet Streaming Protocol) mode for expansion of new services like DOCSIS 3.1.

R-UEPI, the Remote Upstream External PHY Interface, is the upstream interface between the RPD and the CCAP Core. Like R-DEPI, it is an IP pseudowire between the PHY and MAC in an MHAv2 system that contains both a data path for DOCSIS frames, and a control path for setting up, maintaining, and tearing down sessions.

NSI, or the Network Side Interface, is unchanged, and is the physical interface the CMTS uses to connect to the backbone network. Today, this is typically 10 Gbps Ethernet.

CMCI, or Cable Modem to Customer Premise Equipment Interface, is also unchanged, and is typically Ethernet, USB, or WiFi. Within this document, CMCI is referred to as RPD.

5.4 Remote PHY Device Architecture

Figure 4 shows the architecture for an RPD.

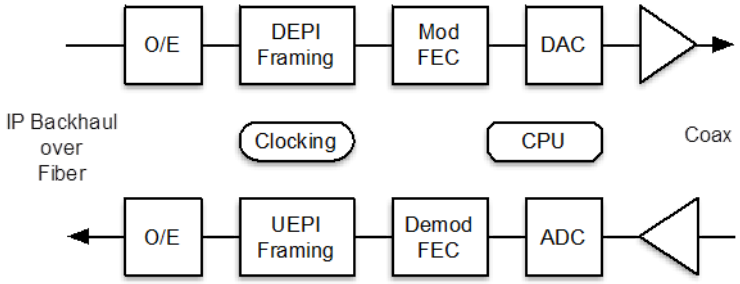

Figure 4 - Remote PHY Device Block Diagram

Packet traffic arrives from the CCAP Core on the downstream receiver. The DEPI framing is terminated; the payload is extracted, framed, modulated, and transmitted out the cable interface. In the upstream, the signal is received from the coax, digitized, demodulated, and the DOCSIS frames are extracted from the FEC payload. The DOCSIS frames are then placed into the UEPI encapsulation and transmitted out the upstream transmitter to the CCAP Core. A clocking circuit interfaces to R-DTI and manages clocking and timing accuracy for the RPD. There is a local CPU that manages the DEPI and GCP control planes and provides an interface into network management.

Figure 4 is meant to be explanatory and is not meant to be all-inclusive. Specific implementations may differ.

5.5 Remote PHY Operation

Figure 5 shows the internal components of an RPD. The following subsections explain the behavior and functionality of these internal components.

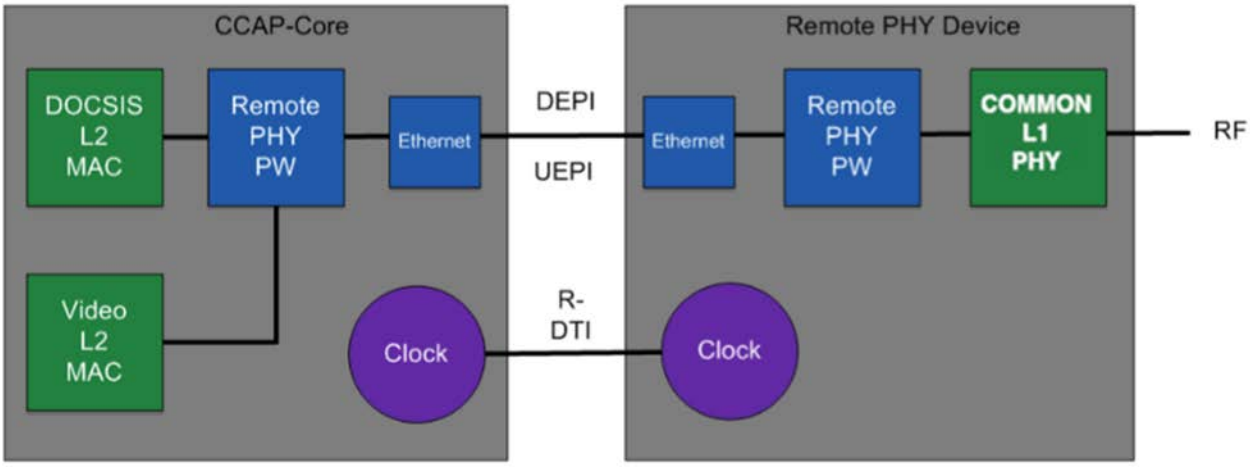

Figure 5 - R-PHY Internal Components[2]

5.5.1 R-DEPI and R-UEPI

R-DEPI and R-UEPI are IP-based pseudowires that are inserted between the DOCSIS MAC in the CCAP Core and the DOCSIS PHY in the RPD. R-UEPI is an extension to R-DEPI. R-UEPI uses the same control plane structure and a unique set of encapsulations in the upstream direction.

[2] Revised per R-PHY-N-16-1584-2 on 9/8/16 by JB.

R-DEPI's job is to take either of the formatted DOCSIS frames, transport them through a Layer 2 or Layer 3 network, and deliver them to the RPD for transmission. R-UEPI's job is to take DOCSIS frames that have been received and demodulated by the DOCSIS upstream PHY in the RPD and transport them to the CCAP Core for processing. The RPD does not provide any upstream DOCSIS processing; with one minor exception, the RPD will extract the bandwidth request frames from the DOCSIS stream and send them in a separate pseudowire so that bandwidth request frames can be given a higher priority than data frames.

The base protocol that is used for the R-DEPI is the Layer 2 Tunneling Protocol version 3, or L2TPv3 for short, and is specified in document [RFC 3931]. L2TPv3 is an IETF protocol that is a generic protocol for creating a pseudowire, which is a mechanism to transparently transport a Layer 2 protocol over a Layer 3 network. Examples of protocols supported by L2TPv3 include ATM, HDLC, Ethernet, Frame Relay, PPP, etc.

Each data packet contains a 32-bit session ID. In the original MPT encapsulation, that session ID is associated with a single QAM Channel. The UDP header-as part of an L2TPv3 encapsulation-is not used in the MHA protocols. The L2TPv3 session ID directly follows the IP header. It is worth noting that the L2TPv3 session ID lands in the same part of a packet as a classic UDP DP/SP. This allows network equipment that classify based upon UDP headers, to be reused for L2TPv3 headers.

L2TPv3 permits creating a subheader whose definition is specific to the payload being carried. The control channel allows for signaling messages to be sent between the CCAP Core and the RPD. Typical control messages will set up a "control connection" between the CCAP Core and the RPD, and then set up multiple data sessions (one for each downstream and upstream QAM or OFDM channel). Each session can be marked with different Differentiated Services Code Points (DSCPs) and can support different encapsulation protocols.

There are two main pseudowire techniques defined by R-DEPI. Each main type supports a variety of subtypes. The first technique, known as MPT mode, transports multiple 188-byte MPEG-TS packets by placing them into the L2TPv3 payload with a unique subheader that contains a sequence number so packet drops can be detected. The encapsulation of DOCSIS frames into MPEG-TS packets is performed in the CCAP Core. The second technique, known as the Packet Streaming Protocol (PSP), transports DOCSIS frames in the L2TPv3 payload. The DOCSIS frames are then encapsulated in MPEG-TS packets within the EQAM. PSP mode allows DOCSIS frames to be both concatenated, to increase network performance, and fragmented, in case the tunneled packets exceed the network MTU size. MPT mode is generally used for single carrier QAM systems such as DOCSIS 3.0 and video, while PSP mode is used for downstream OFDM channels and for the DOCSIS upstream.

5.5.2 Remote DTI

Remote DTI (see [R-DTI]) provides timing synchronization between CCAP Cores and RPDs based on the IEEE 1588v2 standard [IEEE 1588]. The protocol supports the basic synchronization between the CCAP Core and Remote PHY Device for DOCSIS/video/OOB services and the precision time synchronization for emerging services such as wireless backhaul.

5.6 Latency3

One of the technical considerations of the MHAv2 architecture is its impact on the round-trip request-grant delay time. The request-grant delay time is the time from when a CM requests bandwidth, using an uncontended bandwidth request (REQ), to when it receives a MAP message with the granted transmit opportunity in it.

MHAv2 locates the upstream scheduler in the CMTS Core. To prevent the MAP from being slowed down by other traffic in the CIN, the DOCSIS traffic (or a subset containing the MAP messages) may be sent in an independent L2TPv3 flow that can have a unique DSCP. The value of the marked DSCP value should be consistent with a configured "per hop behavior (PHB)" that will provide MAP messages with the highest priority and lowest latency across the CIN to the EQAM. Marking of the DSCP field is optional and part of the operator's overall network design. In the upstream direction, the request can be copied from the DOCSIS frame and sent on an independent L2TPv3 flow that has a unique DSCP.

The net result of prioritizing the MAP and REQ messages, combined with a good CIN design, is to make the operation and performance of the centralized upstream scheduler similar to that of an I-CMTS system.

[3] Revised per R-PHY-N-16.1673-1 on 12/15/16 by JB.

5.7 MHAv2 Summary

In summary, the RPD is used to transfer DOCSIS frames between an IP network interface and an RF interface. The RPD does not participate in the DOCSIS MAC protocol. Instead, MHAv2 provides an IP pseudowire that seamlessly transports the DOCSIS frames between the CCAP Core and the RPD. As such, for most DOCSIS functions, the MHAv2 CCAP system functions almost identically to an I-CCAP. This preserves common functionality and features between the two systems.

6 RPD INITIALIZATION

6.1 Overview

When the RPD device first powers up, it goes through a series of steps before becoming operational. These steps are shown in Figure 6 and explained in this section. Note that Figure 6 is a simplified sequence and does not attempt to show error paths. Error handling during each operation is described in a subsequent, relevant section of the document.

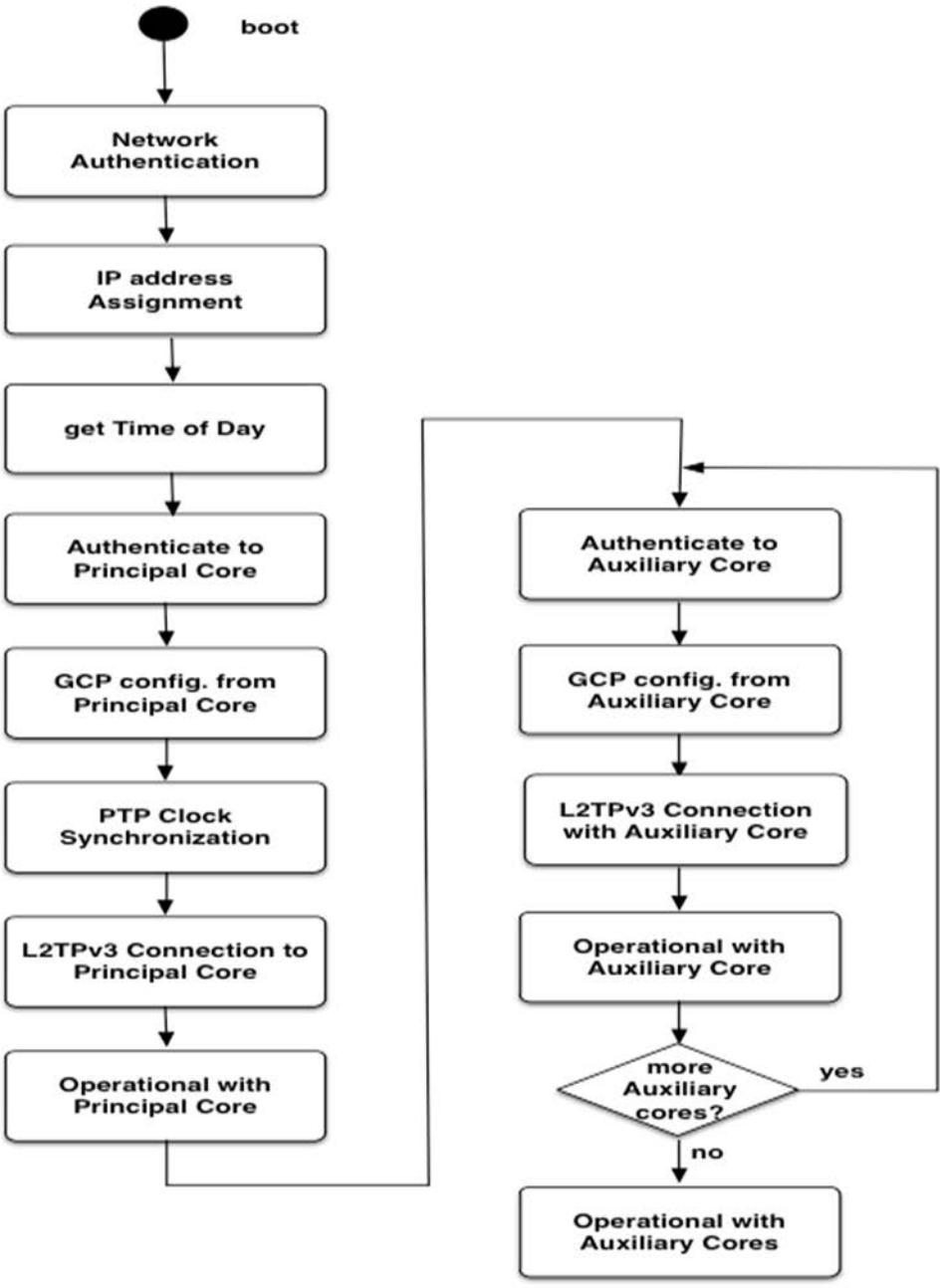

Figure 6 - RPD Initialization

6.2 Security

The Remote PHY security architecture consists of a trusted domain and an untrusted domain (see Figure 7 below). To access the trusted domain and connect to the CCAP Core, RPDs may be required to be authenticated to the trusted network. This is accomplished using 802.1x. When the RPD connects to the CCAP Core, a control session is established which can be secured using IPsec. Both of these mechanisms perform mutual authentication using digital certificate credentials issued from a trusted public key infrastructure (PKI). RPDs support both of these mechanisms (802.1x or IPsec). MSOs can enable them as needed for their specific deployments.

Details for both mechanisms are provided in the following sections.

6.3 Network Authentication

6.3.1 Problem Definition

In many cases, an RPD will be located in an untrusted part of the MSO network, such as a pole-mounted fiber node or remote cabinet but must connect to devices inside the trusted network. When this occurs, it presents a potential security vulnerability. An RPD in an environment like this is shown in Figure 7.

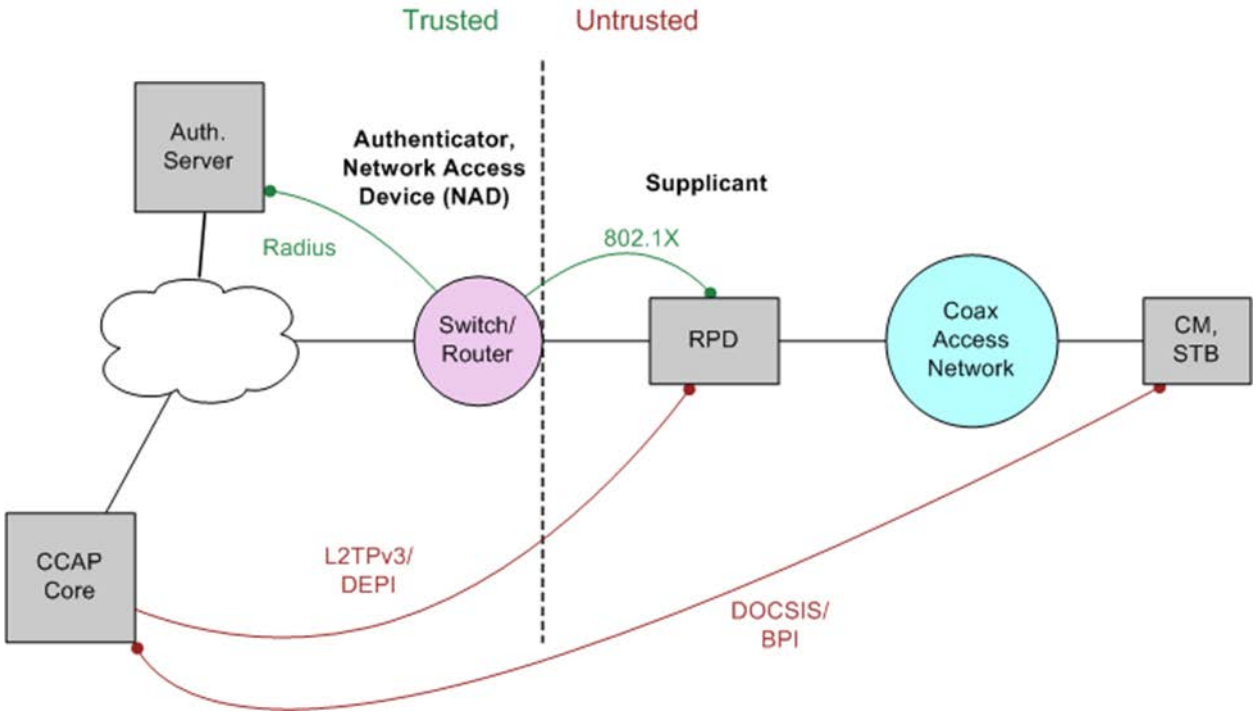

Figure 7 - Remote PHY: Trusted Domain and Untrusted Domain

To mitigate this threat, an MSO may require that the RPD is authenticated before it is allowed access to the trusted network. An RPD may of course be located within the trusted network boundary, such as in a physically secured hub site. In this case, authentication may not be required. Thus, the RPD must be able to operate in both authenticated and unauthenticated networks. Whether authentication is required for an RPD is determined by the network that it is connected to rather than the RPD itself. To support "out of box" operation, an RPD should first attempt to authenticate to the network. If no response to authentication is received, it should assume authentication is not supported by the network and attempt to operate without it (refer to Section 6.3.3.4 for details).

6.3.2 Authentication from an Untrusted Portion of the Network

In Figure 8, the RPD is located in an untrusted area of the network so the network is configured to require authenticated access. The CCAP Core is located in a trusted area of the network. A single RPD may connect to more

than one CCAP Core. This is because there may be different CCAP Cores for DOCSIS and video, or for primary and standby. The RPD will also need to connect to other network services such as DHCP and to allow connections from network management servers.

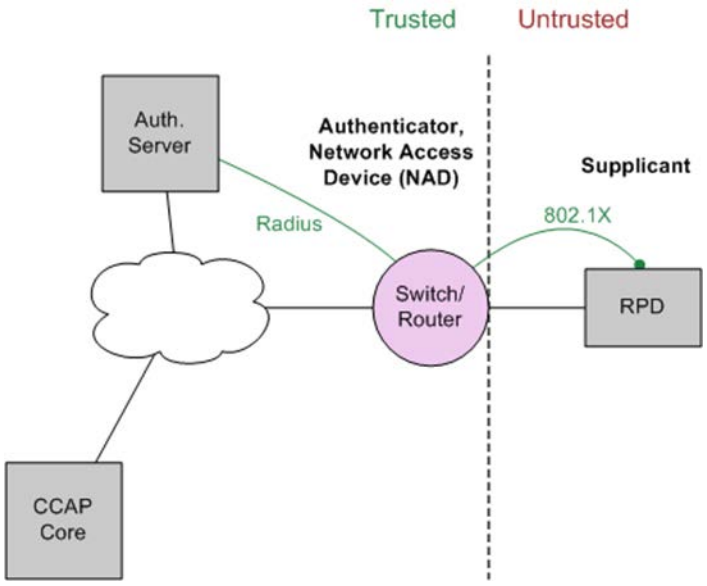

Figure 8 - Authentication Network Diagram

The two authentication scenarios that an RPD MUST support are:

- No authentication in which case the RPD can send to and receive packets from the trusted network with no additional requirements.
- 802.1x based authentication, which requires the RPD to act as an 802.1x supplicant, as described in Section 6.3.3.

A fundamental objective for deployment of RPD is to not require configuration.

To achieve this “out of the box” operation, the RPD MUST be able to determine which security option is in place without configuration.

The RPD MUST determine whether 802.1x authentication is operating, as described in Section 6.3.3.

6.3.3 802.1x Authentication

Authentication is performed based on the 802.1x [IEEE 802.1x] and MACsec [IEEE 802.1ae] standards.

802.1x is a Layer 2 protocol that uses EAP (Extensible Authentication Protocol) to provide authentication services.

For the RPD, EAP-TLS is used based on digital certificate credentials issued from the DOCSIS PKI.

The standard defines three entities:

Supplicant	This is the RPD that requires authentication.
Authenticator/NAD	This is a network element that prevents the RPD from gaining network access until authentication is achieved. The Authenticator is also known as a Network Access Device (NAD).
Authentication Server	This is a standard 802.1x authentication server that validates the authentication.

MHAv2 uses a standard version of the 802.1x protocol with EAP-TLS. This method is referred to as network authentication since the entire authentication process happens between the RPD and a Network Access Device (NAD) without the involvement of the CCAP Core.

Figure 9 shows how the EAP messages between the Authentication Server and the Authenticator are carried over the Radius or Diameter, while the EAP messages from the Authentication Server to the RPD are carried over the combination of Radius/Diameter and 802.1x (EAPoL).

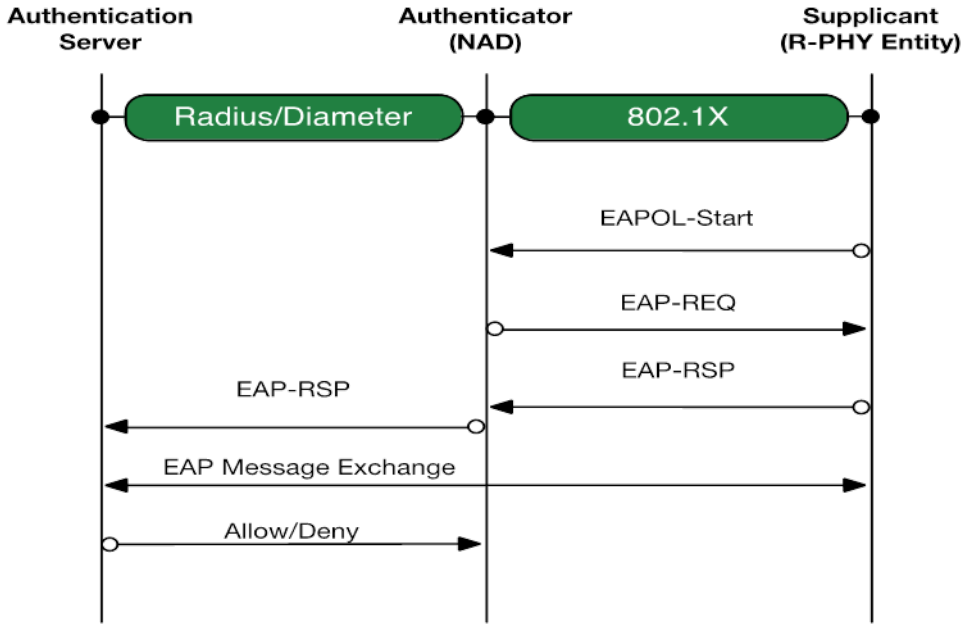

Figure 9 - Network Authentication Signaling

The Authenticator will transmit a Layer 2 broadcast EAP-Request message periodically or in response to an EAPoL start message from a supplicant. An RPD will respond with a Layer 2 unicast EAP-Response. The Authenticator will forward the EAP response to the Authentication server using a RADIUS or DIAMETER protocol. The Authentication server and the RPD then communicate directly using the Authenticator as a relay agent. When the Authentication server has made a decision, it communicates that decision to the Authenticator. The Authenticator will then provide or deny network access to the RPD.

6.3.3.1 MACsec

MACsec (see [IEEE 802.1ae]) is a link layer encryption mechanism used to provide additional security to 802.1x.

MACsec may be used to provide link level encryption between the RPD and the NAD. A security association is created between the NAD and each authenticating RPD based on keying material created during the EAP exchanges. This is used to encrypt data between the NAD and each RPD, providing a higher level of security than basic 802.1x. With 802.1x, after authentication of an RPD, the NAD port is opened to any messages from the authenticated RPD MAC address. This creates the possibility for a device to spoof the RPD MAC address to gain access to the network. With MACsec only devices in possession of legitimate security keys can send traffic to the network.

The use of MACsec provides the following advantages:

- It enables secure access for multiple devices per port
- It provides protection against potential man in middle attacks in both single and multiple devices per port use cases.

MACsec is relatively new and is not yet supported by all switches and all silicon. If MACsec is supported:

- The RPD MUST support the MACsec Key Agreement protocol (MKA) for key exchange and management.
- The RPD MUST derive the Connectivity Association Key (CAK) from the EAP-MSK as defined by EAP-TLS and 802.1x.
- The RPD MUST not use pre-shared CAKs.

6.3.3.2 RPD Topology Support for 802.1x

Figure 10 shows a number of potential topologies for RPD deployment and connectivity to the NAD. The various topologies are discussed in the subsections of this section.

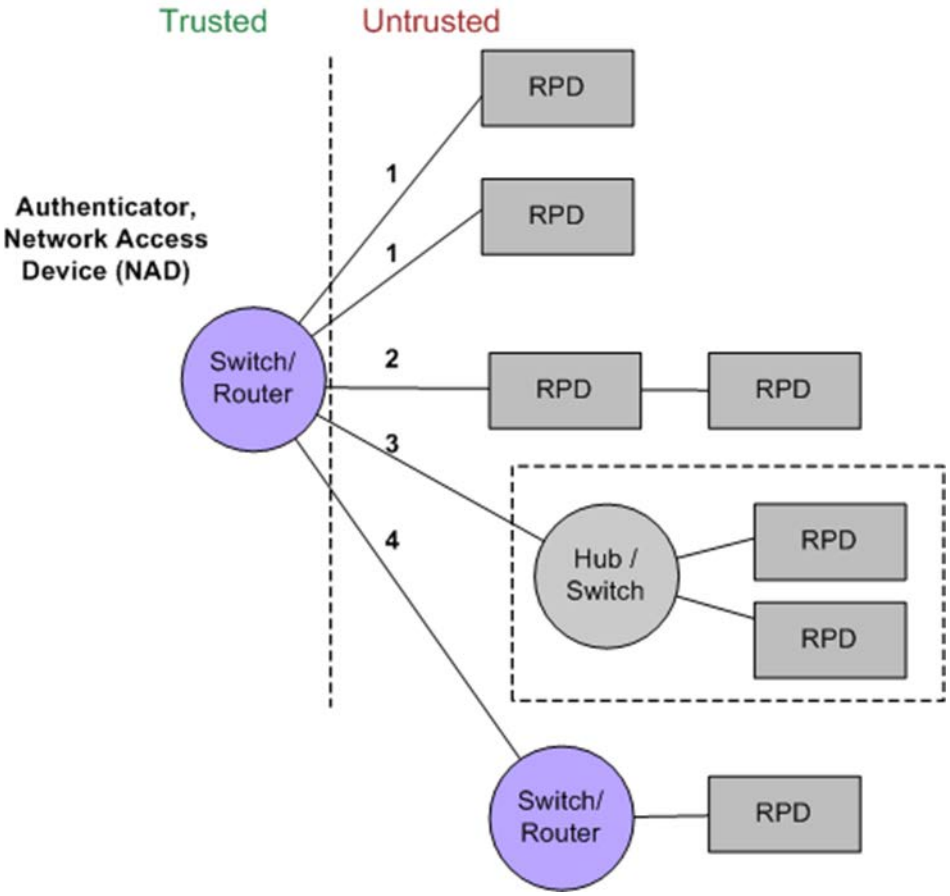

Figure 10 - RPD Topologies for 802.1x

6.3.3.2.1 Type 1: Single Host per Port

In its most basic form, 802.1x supports access by a single host per Ethernet switch port. This is defined in the 802.1x standard and is widely supported by existing switches.

The RPD MUST support this topology.

The RPD MUST support 802.1x in this configuration.

The RPD SHOULD support MACsec in this configuration.

6.3.3.2.2 Type 2: Daisy Chained RPDs

In this topology, a single NAD port is connected to multiple RPDs connecting over a single port. In this topology MACsec may be used to establish independent security associations with each RPD. This is defined in the standard but is not widely supported in current switches.

The RPD MAY support this daisy chain topology.

If a daisy chain topology is supported:

- The RPD SHOULD support 802.1x in this configuration;
- The RPD SHOULD support MACsec in this configuration.
- 802.1x request messages are carried in a multicast packet with the well-known PAE group address as the destination address. Normal Ethernet switches are required to block this address so that 802.1x is typically a single hop protocol with the authenticator directly connected to the supplicant.
- The RPD SHOULD propagate the 802.1x EAP-REQ multicast messages between the NAD and the daisy chain port.

6.3.3.2.3 Type 3: Multiple RPDs in Single Device

In this topology, a single NAD port is connected to an integrated device such as a node with multiple RPDs connecting via an internal hub or switch. Thus the NAD sees multiple devices (and multiple MAC source addresses) on the port. In this topology, MACsec may be used to establish independent security associations with each RPD, based on the RPD MAC address. This is defined in [IEEE 802.1ae], but is not widely supported in current switches.

If this topology is supported:

- Each RPD MUST have a unique MAC address per Ethernet port;
- The internal hub / switch SHOULD propagate the 802.1x EAP-REQ multicast messages to the RPDs;
- The RPD SHOULD support 802.1x in this configuration.
- The RPD SHOULD support MACsec in this configuration.

6.3.3.2.4 Type 4: Intermediate External Switch / Router

In this topology, a one or more RPDs are connected to the NAD through an intermediate switch or router.

MSOs deploying this topology may not be able to utilize 802.1x or MACsec due to the forwarding restrictions on PAE multicast in Ethernet bridges and switches. Definition of the external switch behavior that would be required to support this topology is outside the scope of this specification.

6.3.3.3 Authenticator Location

The Authenticator is hosted in the device at the border of the trusted network. This may be a Layer 2 switch, a Layer 3 router or the CCAP Core.

There can be zero or more Layer 2 switches and/or Layer 3 routers between the Authenticator and the CCAP Core. There can be zero or more Layer 2 switches and/or Layer 3 routers between the Authenticator and the Authentication Server.

6.3.3.4 Operation

After powering up (and prior to obtaining an IP address), the RPD MUST attempt to authenticate itself to the network using 802.1x as shown in Figure 11.

The RPD MUST send an EAPOL-START message to the Authenticator and wait for an EAP-REQ. If there is no EAP-REQ in response to the EOPOL-Start within “EAP-REQ-TIMEOUT”, the EAPOL-START MUST be resent and the RPD MUST return to wait mode. If no EAP-REQ is received after EAPOL-START-RETRIES have been exhausted, the RPD MUST assume that the network is not authenticated, operate in a non-authenticated mode, and proceed with the DHCP phase of the initialization sequence (this is standard operating procedure for an 802.1x device).

If an EAP-REQ is received, the RPD MUST proceed with 802.1x authentication. If the RPD authentication is rejected or if the authentication process fails after an EAP-REQ has been received, the RPD MUST hold off for the defined 802.1x wait period before trying to re-authenticate.

Once authentication is completed successfully, the RPD MUST proceed with the DHCP phase of the initialization sequence.

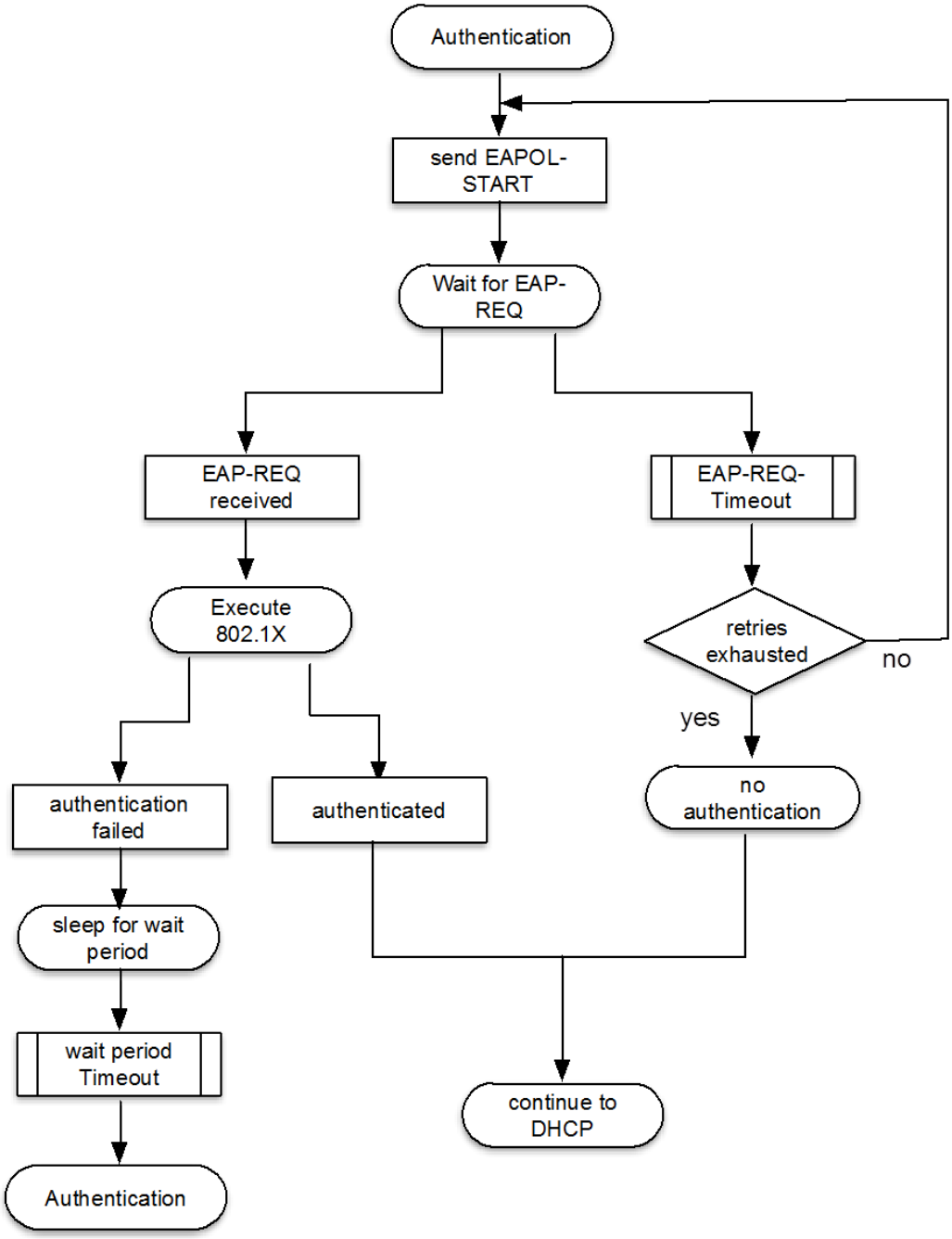

Figure 11 - RPD Authentication using 802.1x

6.3.3.5 802.1x Mutual Authentication

802.1x with EAP-TLS provides mutual authentication of the RPD and the Authentication server. The RPD MUST use EAP-TLS per [RFC 5216] with certificates issued from the DOCSIS PKI managed by CableLabs (see Annex D). The CableLabs Root CA certificate is installed in the Authentication server and RPD as a trust anchor for validating received certificates. The RPD Certificate and its private key, along with the issuing intermediate Device CA certificate are installed in the RPD. The Authentication Server Certificate and its private key, along with the issuing intermediate Service Provider CA certificate, are installed on the Authentication server. During the EAP-TLS message exchange, the RPD and Authentication server will send their device/server certificates and the issuing intermediate CA certificate to each other to be validated against the root CA trust anchor certificate. The RPD and Authentication server MUST use the "Basic Path Validation" procedure defined in [RFC 5280] for validating received certificates.

6.3.3.6 CCAP Core Requirements

The CMTS Core MAY act as a NAD if it is directly connected to the RPD. In this case, it MUST support the 802.1x protocol and act as a relay agent to the Authentication server.

6.3.3.7 Authentication Failures

If an EAP-REQ message has been received indicating that authentication is in effect for the network, any subsequent failures during the authentication process MUST be handled per the [IEEE 802.1x] specification. The wait period timer (which defines the time a device must wait after a failed authentication attempt before another attempt is permitted) SHOULD not be reduced below the 60 second default time.

Retransmission behavior for EAP messages (which are forwarded from the authenticator to the authentication server) MUST follow [RFC 3748].

Constant	Value
EAP-REQ-TIMEOUT	10 sec
EAPOL-START-RETRIES	3

6.4 Address Assignment

Figure 12 shows a simplified access network containing an RPD, a CCAP Core and a DHCP Server.

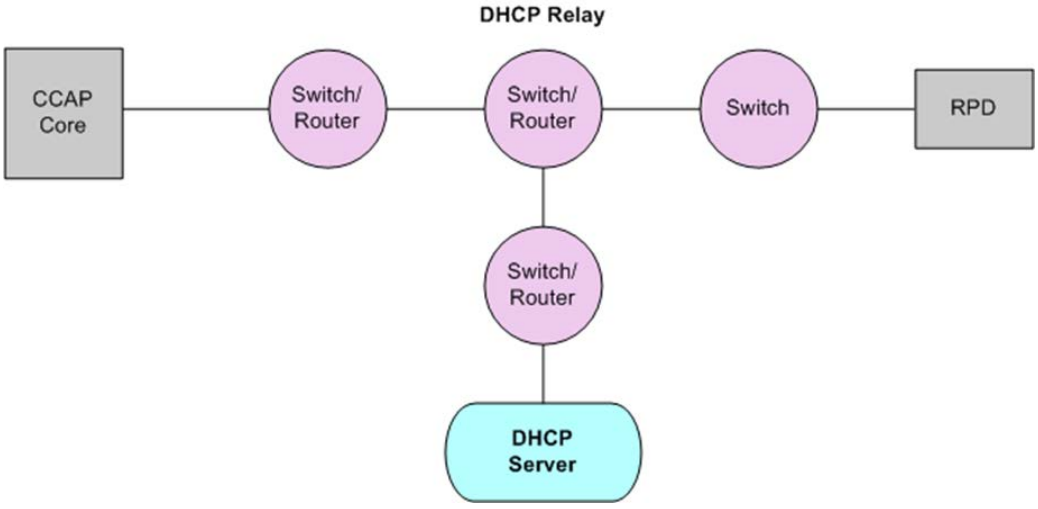

Figure 12 - DHCP Network Diagram

Once the RPD is successfully authenticated to the trusted network, it requests an IP address using DHCP. The standard DHCPv6 or DHCPv4 protocol is used with extensions.

The RPD MUST initially run DHCPv6. If that fails, then the RPD MUST run DHCPv4.

This method is referred to as network DHCP since the entire address assignment of the RPD can take place without the involvement of the CCAP Core. The CCAP Core MAY run a DHCP Relay agent but is not required to if relay is provided by another network component.

There may be zero or more Layer 2 switches between the RPD and the DHCP Relay. The DHCP Relay may be hosted by a Layer 2 switch, a Layer 3 router or the CCAP Core. There may be zero or more Layer 2 switches and/or Layer 3 routers between the network element that is hosting the DHCP Relay and the CCAP Core. There may be zero or more Layer 2 switches and/or Layer 3 routers between the DHCP Relay and the DHCP Server.

Figure 13 shows the DHCP signaling protocol. The RPD issues a broadcast DHCP discovery message when it needs to obtain an IP address. The DHCP agent responds with a unicast DHCP offer that contains the IP address of one or more DHCP servers. The RPD picks one of the DHCP servers and sends a DHCP request to it. The DHCP server sends a DHCP acknowledgement with an IP address for the RPD device.

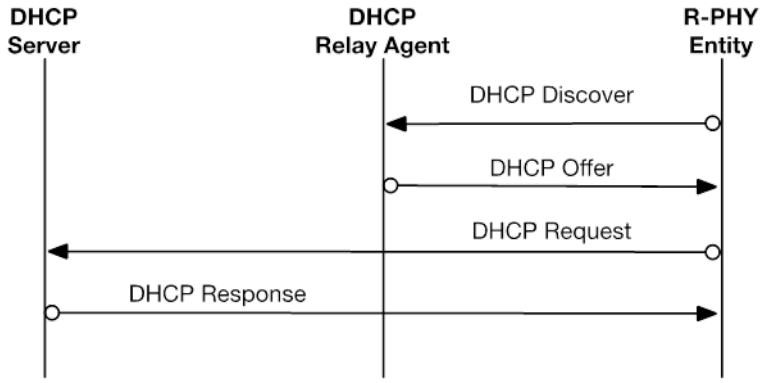

Figure 13 - DHCP Signaling

Unlike the DOCSIS DHCP process where the CCAP is always the DHCP relay agent and can always snoop and append to the DHCP messages, the CCAP Core may not have any direct access to the DHCP message exchange and thus will not be directly aware of the IP address assignment of the RPD. The following mechanism MUST be used to create an association between the CCAP Core and the RPD:

1. The CCAP Core and the DHCP Server MUST be provisioned with the same IP/MAC Address pair for the RPD. The provisioning can be done manually or with an automated system.
2. The DHCP Server MUST provide the IP address of the CCAP Core to the RPD. An additional DHCP option <CCAP Cores> is added to support this mechanism.The CCAP Core MUST either accept the connection from the RPD, deny the connection, or redirect the RPD to another CCAP Core.

6.4.1 DHCP Options[4]

Refer to [CANN] for details on specific options.

The RPD MUST support the following DHCP options when they are received in a DHCP message.

Option	Value	Use
2	Time Offset	Used for authentication, logging, and software upgrade
4	Time Server	Used for authentication, logging, and software upgrade
7	Log Server	Used for logging

[4] Revised per R-PHY-N-16.1637-1 on 12/15/16 by JB.

The RPD MUST support the following CableLabs suboptions under DHCP option 43.

Suboption	Value	Use
2	<Device Type>	MUST be set to “RPD”
3	<ECM: eSAFE>	Not used
4	<serial number>	Refer to [CANN]
5	<hw version>	Refer to [CANN]
6	<sw version>	Refer to [CANN]
7	<Boot ROM version>	Refer to [CANN]
8	<OUI>	Refer to [CANN]
9	<Model Number>	Refer to [CANN]
10	<Vendor Name>	Refer to [CANN]
61	<CCAP Cores>	Address of all CCAP Cores RPD MUST attempt to connect to. The Principal Core is the first entry in the list.

6.4.1.1 CCAP Cores Suboption

CMTS Cores suboption describes either IPv4 or IPv6 addresses, as shown in Figure 14 and Figure 15.

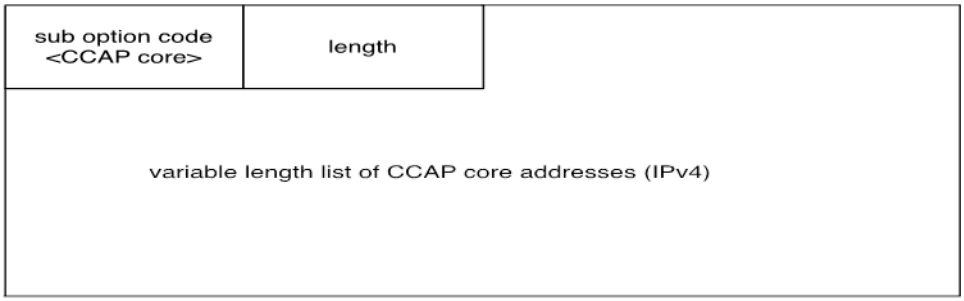

Figure 14 - CCAP Cores DHCP Suboption IPv4

The CMTS Cores suboption can also be used with DHCPv6.

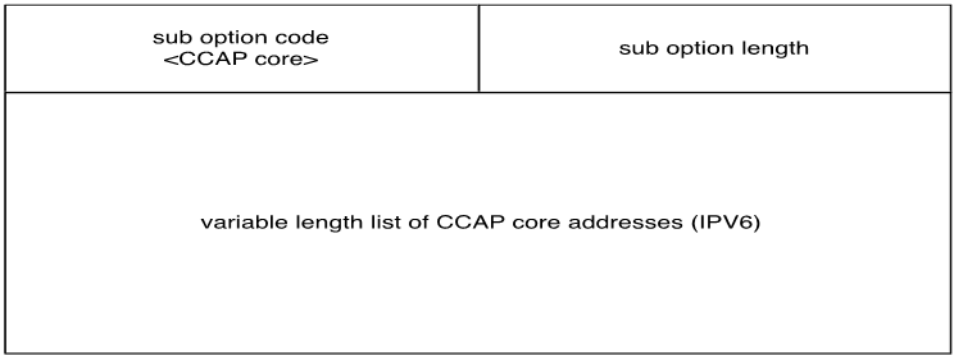

Figure 15 - CCAP Cores DHCP Suboption IPv6

The Cores may have different roles, such as primary, standby, DOCSIS, EQAM, etc. The specific role of each core is determined during the GCP configuration phase.

- The RPD MUST attempt to connect to all cores in the options list as described in Section 6.6.

6.4.2 Failures

The RPD MUST respond to any errors during the DHCP process as per [RFC 2131].

NOTE: This results in the RPD entering a time out and retry loop with a randomized exponential back off.

6.4.3 Security Implications

The RPD MUST attempt to contact the DHCP server via the CIN interface. If 802.1x and MACsec are in place, this will provide secure access to the trusted network.

6.5 Time of Day

The RPD acquires the time of day for the purpose of timestamping warnings, error logs and messages, validation of the CVC during a software upgrade, and validation of the CCAP Core certificate during mutual authentication.

6.5.1 ToD Acquisition

The RPD MUST attempt to obtain the current date and time by using the Time Protocol (see [RFC 868]) from one of the servers listed in the `Time Server Option` DHCP field. If this field is missing or invalid, the RPD MUST initialize the current time to Jan 1, 1970, 0h00. If the time is initialized (reset), the RPD MUST ignore the value, if any, of the Time Offset DHCP option.

The RPD MUST use its DHCP-provided IP address for exchange of messages with the Time Protocol server. The RPD MUST transmit the request using UDP. The RPD MUST listen for the response on the same UDP port as is used to transmit the request. The RPD MUST combine the time retrieved from the server (which is UTC) with the time offset received from the DHCP server to create a valid "local" time.

Once the RPD acquires time, it MUST stop requesting, unless any of its ToD related parameters (such as time offset or server address) are modified. If the RPD's ToD related parameters are modified, the RPD MAY re-request ToD from the Time Protocol server(s).

6.5.2 ToD Conflicts and Problems

The DHCP server may return multiple IP addresses from multiple Time Protocol servers. The RPD MUST attempt to obtain time of day from all the servers listed until it receives a valid response from any of the servers. The RPD MUST contact the servers in batches of tries with each batch consisting of one try per server and each successive try within a batch at most one second later than the previous try and in the order listed by the DHCP message. If the RPD fails to acquire time after any batch of tries, it MUST retry a similar batch using a truncated randomized binary exponential backoff with an initial backoff of 1 second and a maximum backoff of 256 seconds.

If an RPD is unable to establish time of day, it MUST log the failure in the local log. If the RPD does not obtain ToD in the initial request against the first server, the RPD MUST initialize the current time to Jan 1, 1970, 0h00, and then subsequently initialize its current time once it receives a response from a Time Server.

If the RPD fails to establish TOD it will not be able to validate the CCAP Core certificate.

The failure may be due to time server failure or to an error in the DHCP option list.

The RPD will follow the same mechanism as for a DHCP failure and restart the DHCP process as described in Section 6.4.2

6.5.3 ToD Security Implications

The RPD MUST attempt to contact the time server via the CIN-facing port on which the DHCP response was received. If 802.1x and MACsec are in place, this will provide secure access to the trusted network.

6.6 Connection to CCAP Cores

Following successful IP address assignment the RPD attempts to connect to all of the CCAP Cores that have been identified in the DHCP option list as described below.

6.6.1 Core Types

Cores are defined to be either Principal or Auxiliary.

An RPD can be connected to multiple CCAP Cores. Each CCAP Core manages and configures an independent subset of the RPD resources, e.g., one or more RF channels. There are certain types of parameters, which are common across resource sets such as downstream power. The Principal Core is responsible for the configuration of these common parameters for the RPD and for certain device management functions.

Auxiliary Cores are responsible for providing DOCSIS, video, or OOB services. They are restricted to the resource set assigned to them by the Principal Core.

The RPD MUST complete configuration with a Principal Core before allowing configuration from Auxiliary Cores.

In general, it is expected that the first core in the DHCP option list will be the Principal, but the RPD MUST be able to accommodate out-of-order lists.

The RPD MUST accept configuration from only one Principal Core (refer to Section 6.6.2.4 for details).

Principal and Auxiliary Cores may operate in Active or Standby roles.

NOTE: High Availability actions will be defined in a future version of the specification.

6.6.2 Connection Process

The connection process between RPD and each CCAP Core (whether Principal or Auxiliary) consists of two phases:

1. Establish a mutually authenticated secure connection between the RPD and Core.
2. RPD configuration using GCP.

6.6.2.1 Mutual Authentication

Mutual authentication is used when establishing a secure connection between the RPD and CCAP Core(s). It is independent from the authentication used for trusted network access described in Section 6.3.

Mutual authentication is always required between the RPD and CCAP Core but a secure connection may not be required in all cases (e.g., when the RPD is inside the trusted network or MACsec is used to secure access to the trusted network). This is negotiated as described below.

Authentication can be initiated by either the CCAP Core or the RPD.

Whether the RPD is required to authenticate is under control of the CCAP Core.

The RPD and the CCAP Core MUST support mutual authentication based on IKEv2 [RFC 7296] using public key signatures based on the digital certificate credentials issued from the CableLabs DOCSIS PKI (see Annex D). The RPD certificate provisioning requirements are the same as what is defined in Section 6.3.3.5. The CCAP Core MUST be provisioned with the CableLabs Root CA certificate as a trust anchor. The CCAP Core MUST be provisioned with a CCAP Core Device Certificate and its private key along with the CableLabs Device CA certificate which are issued by the CableLabs Root CA.

The RPD MUST use UDP port 500 for IKEv2 exchanges.

The CCAP Core MUST use UDP port 500 for IKEv2 exchanges.

The RPD MUST initiate the IKE_SA_INIT process per [RFC 7296].

The CCAP Core MAY initiate the IKE_SA_INIT process per [RFC 7296] if the RPD address is known in order to facilitate faster reconnection (for example, following a CCAP error).

The RPD MUST attempt to recontact a CCAP Core following a connection loss.

The RPD MUST include its [X.509] certificate in the IKEv2 exchanges.

The CCAP Core MUST include its [X.509] certificate in the IKEv2 exchanges.

The RPD MUST use the value of the common name field from the [X.509] certificate as the identifier in IKEv2 messages it generates.

The CCAP Core MUST use the value of the common name field from the [X.509] certificate as the identifier in IKEv2 messages it generates.

The mechanism by which the CMTS Core determines whether to accept the connection from the RPD is a local matter but could include:

- Local configuration of RPD identifier, or
- Forwarding to an authentication policy server.

The CCAP Core MUST determine the security profile for the GCP control plane during the IKEv2 exchange.

The GCP control plane SHOULD be authenticated by the CCAP Core. The CCAP Core MAY encrypt the GCP Control Plane.

If authentication or encryption are in operation, IPsec ESP in transport mode MUST be used to protect the GCP control plane.

If authentication or encryption are in operation, IKEv2 MUST be used to generate the keying material required to secure the GCP control plane.

The IKEv2 traffic selector MUST be the 5-tuple identifying the GCP connection.

The CCAP Core MUST determine the security profile for the L2TPv3 control plane during the IKEv2 exchange.

The L2TPv3 control plane MAY be authenticated and MAY be encrypted by the CCAP Core.

If authentication or encryption are in operation, IPsec ESP in transport mode MUST be used to protect the L2TPv3 control plane.

If authentication or encryption are in operation, IKEv2 MUST be used to generate keying material to secure the L2TPv3 control plane.

The IKEv2 traffic selector MUST be the 5-tuple identifying the L2TPv3 connection.

Different security associations MUST be used for GCP and L2TPv3 control.

The following cryptographic methods as defined in [RFC 4307] MUST be supported:

- Message integrity using HMAC-SHA1-96;
- Data Encryption using AES 128 CBC;
- Pseudo-random function for key generation HMAC-SHA1;
- Certificate authentication using RSA Signature Algorithm [RSA 3] with SHA-256 hash (see [FIPS 180-4]) per Annex D

6.6.2.1.1 No Authentication Option

If the CCAP Core does not wish to use encryption or authentication, it signals this by selecting null encryption and no authentication options (see [RFC 4307]) during the IKEv2 exchange.

6.6.2.1.2 Certificate Validation

The RPD MUST use the “Basic Path Validation” procedure defined in [RFC 5280] for validating received certificates.

The CCAP Core MUST use the “Basic Path Validation” procedure defined in [RFC 5280] for validating received certificates.

6.6.2.1.3 Authentication Failure

Failures during the authentication process MUST be handled per [RFC 7296].

If the authentication is terminated due to a failure, the RPD will attempt to connect to the next core in the list.

6.6.2.2 RPD Configuration via GCP

Following authentication, the CCAP Core MUST configure the RPD using the GCP (Generic Control Plane) protocol (see [GCP]). Since the RPD is an extension of the CCAP Core, the CCAP Core contains all the necessary configuration information.

GCP allows control plane data structures from other protocols to be tunneled through its generic control plane. For example, GCP can directly use DOCSIS TLVs for the configuration of the RPD PHY parameters.

Note that the [R-DEPI] and [R-UEPI] protocols also contain a certain amount of configuration information. The MHAv2 paradigm is to keep the R-DEPI and R-UEPI configuration focused on session signaling and to use GCP for RPD-specific configuration and operation.

The specific RPD configuration parameters used in GCP are listed in Annex B.

The GCP protocol is authenticated and secured using IPsec. Encryption and/or message authentication codes (HMAC) can be applied to protect packets. IPsec keys are derived from the keying material created during the IKEv2 authentication process. IPsec session key exchange and renewal during the life of the GCP connection will be supported using IKEv2.

6.6.2.3 GCP Connection Failures

If a CCAP Core has not responded after CORE_CONNECT_TIMEOUT, then the RPD MUST retry the connection CONFIG_RETRY_COUNT. If no response is received after retries are exhausted, the RPD MUST move on to the next Core in the list.

Constant	Value
CORE_CONNECT_TIMEOUT	5 seconds
CONFIG_RETRY_COUNT	3

6.6.2.4 Connection to Principal Core[5]

Following successful IP address assignment, the RPD MUST follow the process shown in Figure 16 to connect to a Principal Core.

[5] Revised per R-PHY-N-15.1403-3 on 1/8/16 by JB.

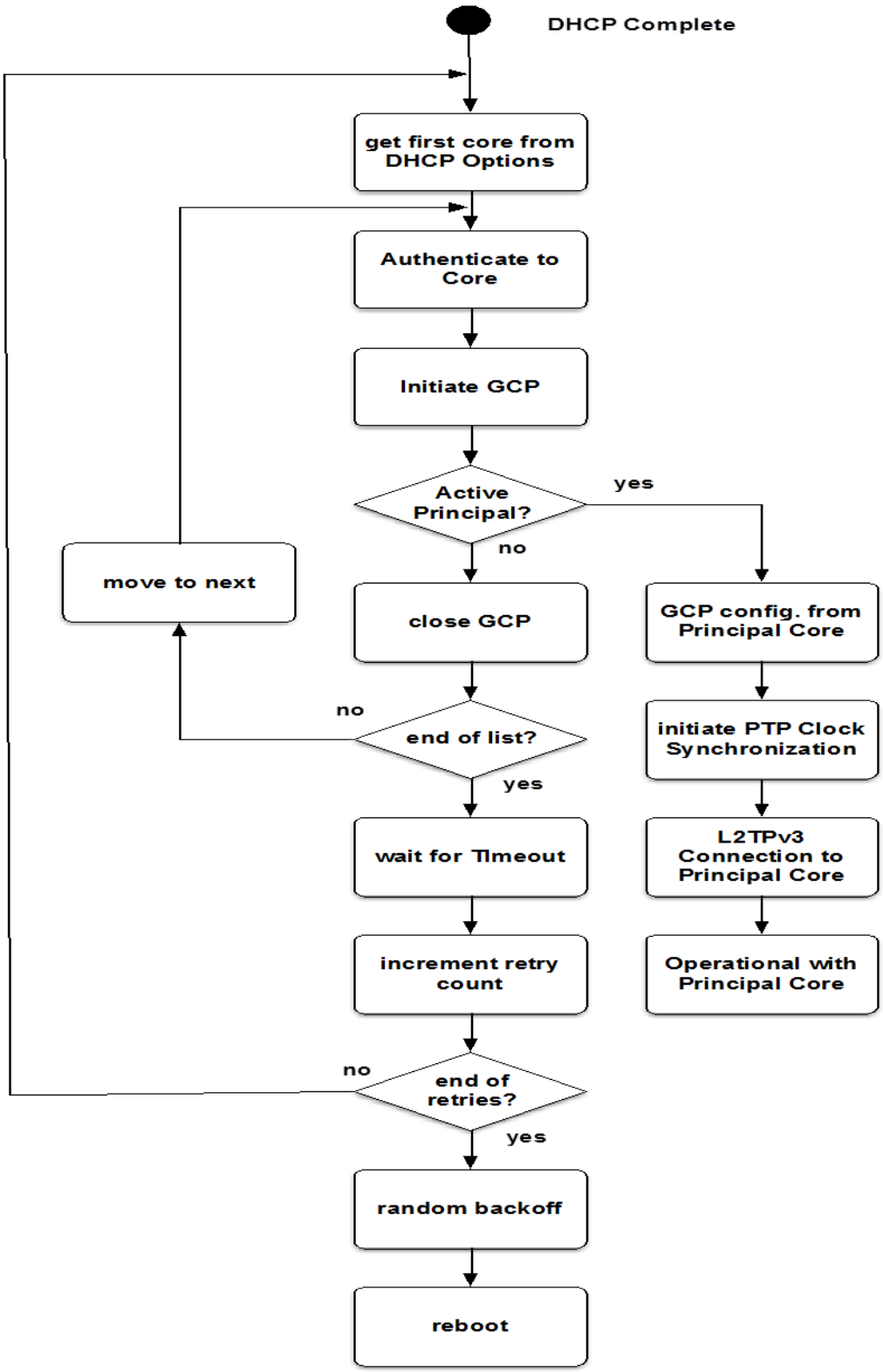

Figure 16 - Process for Connecting to the Principal Core

The RPD MUST establish a GCP connection with a Principal Core following the process shown in Figure 16 and described below. It MUST start the process with the first core in the DHCP option list and move sequentially through the list until a successful connection is achieved.

The RPD MUST attempt to authenticate with the core as described in Section 6.6.2.1.

Following authentication, the RPD MUST initiate a TCP connection to the GCP well-known port on the CCAP Core. When the connection is established, the RPD MUST issue a GCP Notify message to the CCAP Core to identify itself to the CCAP Core. Following the Notify message the CCAP-Core will initiate RPD configuration and determine if the core can act as a Principal.

If the core identifies as a Principal Core in active mode (as opposed to standby), the RPD MUST proceed with GCP configuration.

If the configuration received from the Principal Core overwrites any of parameter values communicated via the DHCP Options previously received, the RPD MUST use the parameter values received from the Principal Core. Alternatively, a new list of Auxiliary Cores could be provided.

When configuration by the Principal Core is complete, the RPD MUST initiate PTP clock synchronization. The PTP may take some time to reach a steady state, so the RPD should start the synchronization process as soon as possible.

The RPD MUST establish L2TPv3 connectivity with the Principal Core as described in Section 6.8.

Following successful L2TPv3 establishment, the RPD is operational with the Principal Core and MUST move on to any Auxiliary Cores defined by DHCP Options or defined during configuration from the Principal Core.

6.6.2.5 Failures[6]

If the core contacted is not a Principal Core, the RPD MUST move to the next core in the options list and attempt to contact this core.

If the end of the option list is reached with no Principal Core found, the RPD MUST wait NO_PRINCIPAL_CORE_FOUND_TIMEOUT then retry from the start of the list.

If no Principal Core can be contacted after PRINCIPAL_CORE_RETRY_COUNT attempts, the RPD MUST wait for a random interval and then reboot. The delay before the first reboot MUST be PC_BACKOFF_MIN randomized by the value of a uniform number chosen from the range -1 to +1. The reboot delay MUST be doubled for subsequent retransmissions up to a maximum of PC_BACKOFF_MAX. If no Principal Core can be contacted after PRINCIPAL_CORE_RETRY_COUNT attempts, the RPD MUST wait for a random backoff time between PC_BACKOFF_MIN and PC_BACKOFF_MAX, and then reboot.

Constant	Value
NO PRINCIPAL_CORE_FOUND_TIMEOUT	60 seconds
PRINCIPAL_CORE_RETRY_COUNT	3
PC_BACKOFF_MIN	60 seconds
PC_BACKOFF_MAX	300 seconds

6.6.2.6 Redirection[7]

If a Principal CCAP Core does not have configuration data for an RPD or is not aware of the RPD, the core SHOULD either reject the connection and log an error or use GCP to redirect the RPD to another core.

A CCAP Core MAY elect to redirect an RPD to one or more alternate CCAP Cores for further configuration, e.g. , to act as a standby or to provide additional services.

The CCAP Core MUST use the GCP (Generic Control Plane) protocol to redirect the RPD.

[6] Revised per R-PHY-N-15.1357-2 by JB on 9/21/15.
[7] Revised per R-PHY-N-15.1403-3 on 1/8/16 by JB. Revised per R-PHY-N-16.1575-1 on 9/19/16 by JB.

The redirecting CCAP Core MUST transfer a variable length list of IPv4 or IPv6 addresses to the RPD. The redirecting CCAP Core MAY delay providing the redirect information to the RPD for a period of up to 60 seconds.

6.6.3 Connection to Auxiliary Cores

After becoming operational with a Principal Core, the RPD MUST follow the process shown in Figure 17 to connect to any Auxiliary Cores that have been configured.

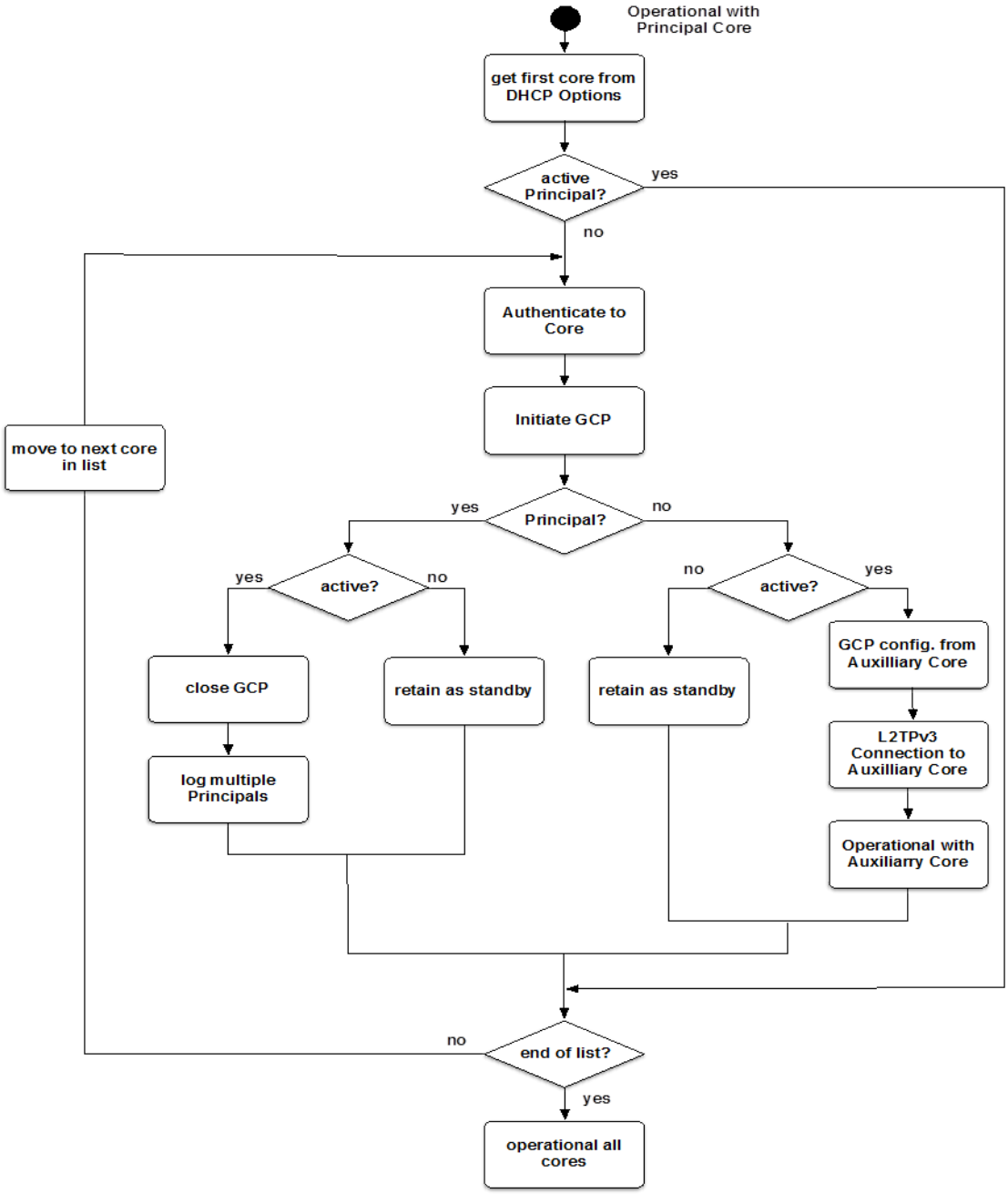

Figure 17 - Process for Connecting to Auxiliary Cores

For each core in the list:

- If this is the active Principal Core (to which it is already connected) the RPD MUST move to the next entry in the list.
- The RPD MUST try to authenticate with the core.
- Following authentication the RPD MUST initiate a TCP connection to the GCP well-known port on the CCAP Core. When the connection is established, the RPD MUST issue a GCP Notify message to the CCAP Core to initiate configuration.
- If the core is an additional Principal Core operating in active mode, the RPD MUST log an error and close the GCP connection because only one active Principal Core is allowed. The CCAP Core MUST also log an error.
- If the core is a Principal Core operating in standby mode the RPD MUST retain this information in case the active Principal Core fails.
- If the core is an auxiliary operating in standby mode the RPD MUST retain this information in case the active Auxiliary Core fails.
- If the core is an auxiliary core operating in active mode:
    - The RPD MUST proceed with GCP configuration.
    - The RPD MUST establish L2TPv3 connectivity with the Auxiliary Core as described in Section 6.8.
    - Following successful L2TPv3 establishment, the RPD is operational with the Auxiliary Core and MUST move on to any additional Auxiliary Cores defined by DHCP Options or defined during configuration from the Principal Core.

6.6.4 Reboot Hold[8]

If an RPD is having trouble booting it may be necessary to use an SSH or local console session to debug the problem. To prevent the reboot operation from disrupting the process the RPD MUST provide a control variable (PC_REBOOT_HOLD) that can be set from the CLI during the debug session to prevent a reboot.

While PC_REBOOT_HOLD is TRUE the RPD MUST NOT reboot.

PC_REBOOT_HOLD MUST be set to FALSE when the RPD boots so that a power cycle overrides the debug hold.

The CLI MUST provide commands to set and clear PC_REBOOT_HOLD.

6.7 Synchronization

Once the RPD has been configured, the RPD chooses its method of synchronization. The RPD can be directed to either be internally synchronized where the RPD is the clock master (Option A) or externally synchronized where the RPD is a clock slave (Option B).

In a Remote PHY system, the downstream and upstream PHY timing are always aligned because the downstream and upstream PHY are co-located. The only timing requirement is to be able to share a timestamp value between the CCAP Core and the RPD for upstream scheduling. These timing techniques are described in [R-DTI].

Note that if the upstream scheduler is located in the RPD, then all the timing elements are local to the RPD and no adjustments are necessary. This scenario is equivalent (from a timing standpoint) to having the entire CMTS in the RPD. This is a future option for the R-PHY architecture should it ever be needed.

The net effect of all methods is that the timestamp used in the SYNC message, the MAP message, the REQ message, the RPD upstream burst receiver, and the upstream scheduler are aligned.

The protocol used in [R-DTI] between the CMTS Core and the RPD is the Precision Time Protocol (PTP) as defined by [IEEE 1588]. PTP is used because it is a standard protocol whose accuracy can be enhanced when the CIN is

[8] Added per R-PHY-N-15.1357-2 by JB on 9/21/15.

built with [IEEE 1588] compliant equipment. Note that it is not necessary for the network to be compliant to [IEEE 1588].

Encryption or authentication of PTP messages between the master clock and the RPD (e.g., by using IPsec) would result in some loss of accuracy because intermediate nodes could not update the timing data. If security of PTP messages is required, MACsec encryption can be used.

The synchronization requirements can be summarized as follows:

- All specific operational requirements are stated in the [R-DTI] specification.
- The RPD MUST be able to support PTP messages received over a MACsec (see [IEEE 802.1ae]) secured link from the CIN.
- If the CCAP Core has specified a PTP Master source during GCP configuration the RPD MUST use it.

6.7.1 Synchronization Failures

6.7.1.1 RPD Operating as a Timing Slave

If the RPD does not receive a sync message within PTP_SYNC_TIMEOUT, it MUST log a LOSS_OF_SYNC error and operate in a non-synchronized local clocking mode. It will continue to attempt to synchronize with a PTP clock master. When synchronization is re-established, a SYNC_ESTABLISHED message MUST be logged.

6.7.1.2 RPD Operating as a Timing Master

If the RPD does not receive a delay request message within PTP_DELAY_TIMEOUT, it MUST log a LOSS_OF_SLAVE error. It will continue to act as PTP clock master. When communication with a slave is re-established, a SLAVE_FOUND message MUST be logged.

Constant	Value
PTP_SYNC_TIMEOUT	5 seconds
PTP_DELAY_TIMEOUT	5 seconds

6.8 Connectivity[9]

Once clocking has been established, the RPD and the CCAP Core are ready to set up the L2TPv3 data tunnel and control plane connectivity.

Downstream data plane connectivity between the CCAP Core and the RPD is described in [R-DEPI]. Upstream data plane connectivity between the RPD and the CCAP Core is described in [R-UEPI]. [R-DEPI] is an extension of the Layer 2 Tunneling Protocol described in [RFC 3931].

R-DEPI and R-UEPI establish one overall control plane connection between a CCAP Core and an RPD pair. Within this tunnel, there are separate pseudowires consisting of L2TPv3 sessions for each MAC-PHY functional pair. The data plane encapsulation is managed per session. Depending upon the type of pseudowire encapsulation used, a pseudowire may contain one or more channels.

After the R-DEPI and R-UEPI session initializes, the CCAP Core and RPD are ready to use.

The connectivity requirements can be summarized as follows:

- If the RPD is provided with a Principal CCAP Core IP address, the RPD MUST try to establish connectivity to that CCAP Core using the primary R-DEPI session.
- If the RPD is provided with an Auxiliary CCAP Core IP address, the RPD MUST try to establish connectivity to that CCAP Core using the auxiliary R-DEPI session.
- The CCAP Core MUST be compliant with [R-DEPI] and [R-UEPI].

[9] Revised per R-PHY-N-16.1551-1 on 8/10/16 by JB.

- The RPD MUST be compliant with [R-DEPI] and [R-UEPI].
- The CCAP Core MAY support the R-DEPI D-MPT or MCM pseudowire types for transport of data on DOCSIS SC-QAM channels.
- The CCAP Core MUST support the R-DEPI D-MPT pseudowire type for MPEG-TS video packets.
- The CCAP Core MAY support the R-DEPI MCM pseudowire type for MPEG-TS video packets.
- The CCAP Core MUST support all R-DEPI PSP pseudowire types for transport of data on DOCSIS SC-QAM and OFDM channels.
- The CCAP Core MUST support all R-UEPI PSP pseudowire types.
- The RPD MAY support the R-DEPI D-MPT or MCM pseudowire types for transport of data on DOCSIS SC-QAM channels.
- The RPD MUST support the R-DEPI D-MPT pseudowire type for MPEG-TS video packets.
- The RPD MAY support the R-DEPI MCM pseudowire type for MPEG-TS video packets.
- The RPD MUST support all R-DEPI PSP pseudowire types for transport of data on DOCSIS SC-QAM and OFDM channels.
- The RPD MUST support all R-UEPI PSP pseudowire types.

7 SECURE SOFTWARE DOWNLOAD

7.1 Introduction[10]

Remote PHY architecture supports downloading code to RPDs. Authenticating the source and verifying the integrity of downloaded code is vital to the overall operation and security of Remote PHY architecture. The methods for secure software download as well as the relevant specification text have been adopted from the DOCSIS 3.1 Security Specification [SECv3.1]. The secure software download (SSD) functionality is generally applicable to Remote PHY devices installed in unsecure locations.

Broadly speaking, with respect to secure software downloads; the RPD assumes the functions of a DOCSIS cable modem. It is envisioned that such an approach will allow the operators to reuse the majority of the OSS infrastructure deployed for CM software and security certificate management to perform equivalent functions for RPDs. However, there are important changes to the upgrade procedure. These changes are summarized below and explained further within Section 7.

- The RPD upgrade process relies on certificates from the new CableLabs PKI. The legacy certificates are not supported.
- Unlike a DOCSIS CM, the RPD does not receive a configuration file from a provisioning system. RPD initialization involves connecting to and obtaining configuration information from a Principal CCAP Core via GCP. The software upgrade TLVs received via GCP effectively replace equivalent TLVs received by a CM in a configuration file.
- Unlike a CM SSD process, which needs to be enabled by inclusion of CVC in the CM configuration file, the RPD is implicitly enabled for SSD. The Principal CCAP Core maintains control over this feature because it has control over GCP.
- A DOCSIS CM receives time service from the provisioning system via the DOCSIS Time Protocol, while an active RPD is time synchronized via the PTP protocol. The RPD needs to receive time service via the DOCSIS Time Protocol before it establishes the PTP protocol connection.

The RPD code is signed with a certificate from the new PKI defined in [SECv3.1] and then validated by the RPD. The software download module is an attractive target for an attacker. If an attacker were able to mount an attack against the software download module, s/he could potentially install code to disrupt service on a wide scale or to redirect the content. To thwart these attacks, the attacker is forced to overcome several security barriers.

7.2 Overview[11]

The requirements defined in this section address these security objectives for the code download process:

- The RPD needs to have a means to authenticate that the originator of any download code is a known and trusted source;
- The RPD needs to have a means to verify that the downloaded code has not been altered from the original form in which it was provided by the trusted source;
- The process needs to simplify the operator's code file-handling requirements and provide mechanisms for the operator to upgrade or downgrade the code version of RPDs on their network;
- The process allows operators to dictate and control their policies with respect to: 1) which code files will be accepted by RPDs within their network; and 2) security controls that establish the security of the process on their network;
- RPDs are able to move freely among systems controlled by different operators;
- Support updating the Root CA Certificate in the RPD (optional);
- Support updating the Device CA Certificate in the RPD (optional).

[10] Revised per R-PHY-N-15.1410-2 on 1/11/16 by JB.
[11] Revised per R-PHY-N-15.1410-2 on 1/11/16 by JB.

The concerns of individual operators or RPD manufacturers may result in additional security related to the distribution or installation of code into a RPD. This specification does not restrict the use of further protections, as long as they do not conflict with the requirements of this specification.

Multiple levels of protection are required to protect and verify the code download:

- The manufacturer of the RPD code always applies a digital signature to the code file. The signature is verified with a certificate chain that extends up to the Root CA before accepting a code file. The manufacturer signature affirms the source and integrity of the code file to the RPD;
- Though the manufacturer always signs its code file, an operator may later apply its code signature in addition to the manufacturer signature. If a second signature is present, the RPD verifies both signatures with a certificate chain that extends up to the Root CA before accepting a code file;
- OSS mechanisms for the provisioning and control of the RPD are critical to the proper execution of this process. SSDs are initiated by the Principal CCAP-Core during the initial RPD configuration process, or during normal operation. The operator controls this process indirectly via CLI or SNMP interface on the Principal CCAP-Core. The RPD SSD can be also initiated by operators via SSH and CLI directly on the RPD.

The RPD code file is built using a [PKCS#7]-compliant structure that is defined below, which is identical to the code structure used to upgrade CM software. Included in this structure are:

- The upgrade code image;
- The Code Verification Signature (CVS); i.e., the digital signature over the code image and any other authenticated attributes as defined in the structure;
- The Code Verification Certificate (CVC); i.e., an [X.509]-compliant certificate that is used to deliver and validate the public code verification key that will verify the signature over the code image. The DOCSIS Certificate Authority (CA), a trusted party whose public key is already stored in the RPD, signs this certificate.

Figure 18 shows the basic steps required for the signing of a code image when the code file is signed only by the RPD manufacturer, and when the code file is signed by the RPD manufacturer *and* co-signed by an operator.

In DOCSIS, the Root CA certificate is installed in each RPD as a trust anchor. The code manufacturer builds the code file by signing the code image using a DOCSIS [PKCS#7] digital signature structure with a Manufacturer CVC certificate and the issuing CVC CA certificate. The code file is then sent to the operator. The operator verifies that the code file is from a trusted DOCSIS manufacturer and has not been modified. At this point, the operator has the option of loading the code file on the Software Download server as-is, or of adding its signature and operator CVC and issuing CVC CA certificate to the code file. During the code upgrade process, the RPD retrieves the code file from the Software Download server and verifies the new code image using the Root CA Certificate trust anchor before installing it. See Annex D for CVC chain details.

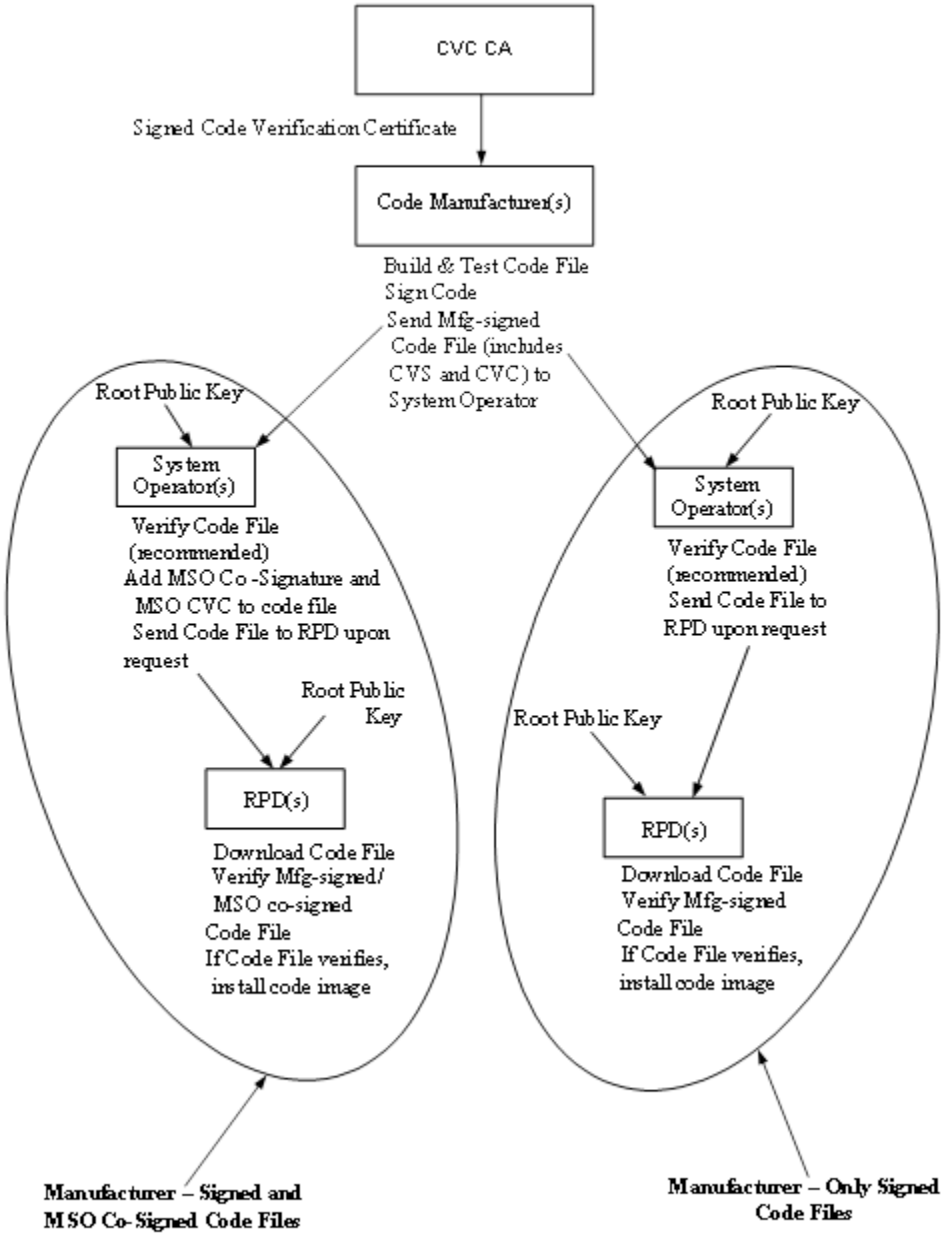

Figure 18 - Typical Code Validation Hierarchy

7.3 RPD Software Upgrade Procedure[12]

This section outlines RPD software upgrade procedure intended to enable automation of SW upgrades. The RPD SW upgrade procedure is presented in Figure 19.

[12] Revised per R-PHY-N-15.1410-2 on 1/11/16 by JB.

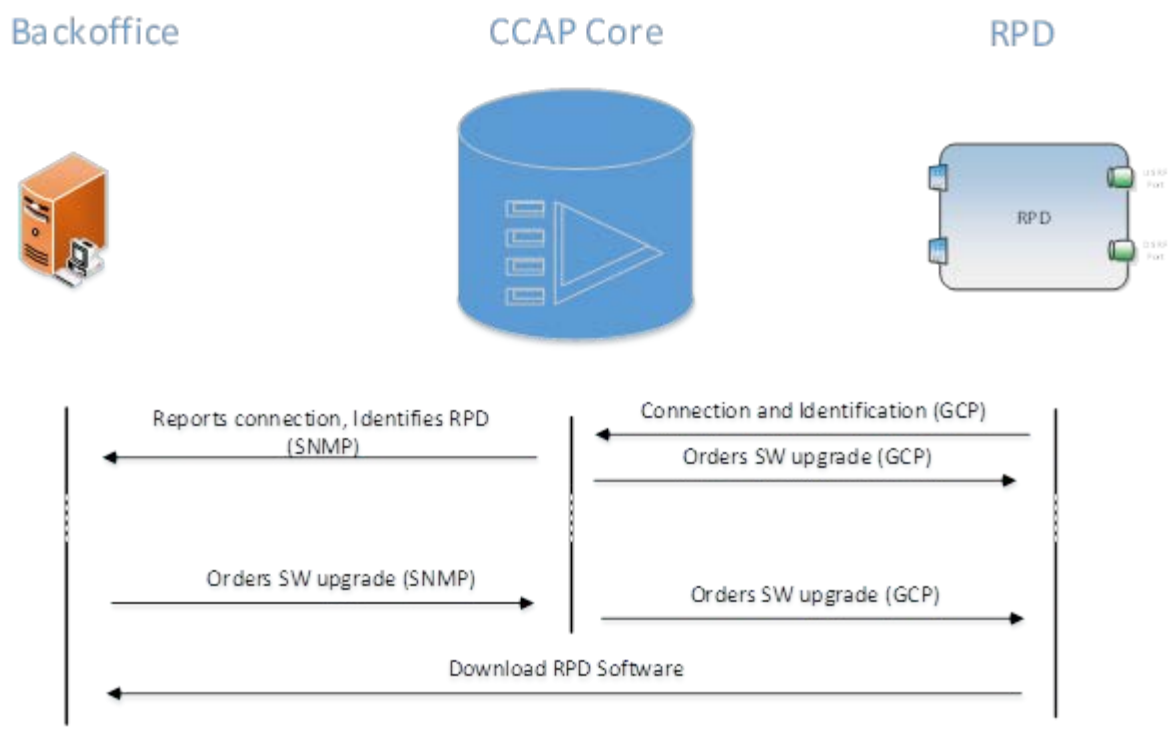

Figure 19 - RPD SW Upgrade Procedure

During the initialization, after the RPD is network authenticated (or bypasses network authentication) and has obtained an IP address, it authenticates to the Principal CCAP-Core and establishes a GCP connection. In the next step the RPD identifies itself to the Principal CCAP Core via GCP. Once the CCAP has identified the RPD and accepts the connection, it will report the RPD to MSO BackOffice system via Syslog or SNMP trap. At that point the CCAP Core can command the RPD to perform the SW upgrade or proceed with RPD configuration. The SW upgrade during the connection initialization is necessary to permit critical SW upgrades in case of incompatibility that may prevent successful pairing of the RPD and the CCAP-Core.

The CCAP Core can be instructed via SNMP or CLI to order the RPD to perform the software upgrade at any time.

In both cases, from the perspective of the RPD the software upgrade is initiated by Principal CCAP-Core via GCP software update option.

The operator may also connect to the RPD directly via SSH, provide necessary SSD parameters and command the RPD to update its software. This option is available even before the RPD is connected to any CCAP-Core device. The detailed description of this method is outside of the scope of this specification.

The RPD MUST support a software download initiated through the GCP software update feature. The RPD MUST support a secure software download initiated via SSH and CLI directly on the RPD.

A software update is accomplished by providing the RPD with a set of parameters which include a filename, an IP address (v4 or v6) for a software download server and Manufacturer's and Co-signer Code Validation Certificates (CVCs). The RPD then uses TFTP or HTTP to go to that server to retrieve the software update file.

NOTE: This method of software update is intentionally similar to how a DOCSIS CM is assigned a new software image so that the existing DOCSIS infrastructure may be leveraged.

The RPD MUST implement a TFTP client compliant with [RFC 1350] for software file downloads. The RPD MUST implement an HTTP-client compliant with [RFC 1945] or [RFC 2616] for software file downloads. The transfer is initiated by the CCAP Core via GCP, as described here.

The RPD MUST include the TFTP block size option [RFC 2348] when requesting the software image file.

The RPD MUST request a block size of 1448 octets if using TFTP over IPv4.

The RPD MUST request a block size of 1428 octets if using TFTP over IPv6.

If the file specified in the GCP Software Upgrade File Name TLV does not match the current software image of the RPD, the RPD MUST request the specified file via TFTP from the software server. The RPD selects the software download server as follows:

- If the RPD communicates with CCAP Core via IPv4 and receives the Software Upgrade IPv4 TFTP Server TLV via GCP, the RPD MUST use the server specified by this TLV. The RPD MUST ignore the Software Upgrade IPv6 TFTP Server TLV when it communicates with CCAP Core via using IPv4.
- If the RPD communicates with CCAP Core via IPv6 and receives the Software Upgrade IPv6 TFTP Server TLV via GCP, the RPD MUST use the server specified by this TLV. The RPD MUST ignore the Software Upgrade IPv4 TFTP Server TLV when it communicates with CCAP Core via IPv6.

When performing a GCP-initiated software download, the RPD MAY defer normal operation until the download is complete. The RPD MUST verify that the downloaded image is appropriate for itself. If the image is appropriate, the RPD MUST write the new software image to non-volatile storage. Once the file transfer is completed successfully, and the software image is verified, the RPD MUST restart itself with the new code image with a RPD Initialization Reason of SW_UPGRADE_REBOOT.

If the RPD is unable to complete the file transfer for any reason, it MUST remain capable of accepting new software download requests (without operator or user interaction), even if power or connectivity is interrupted between attempts. The RPD MUST log the failure. The RPD MAY report the failure asynchronously to the network manager. The RPD MUST continue to operate with the existing software if an upgrade cannot be performed.

If the RPD receives a valid image, it will automatically upgrade its software, reboot and repeat the entire initialization process, including authentication. Image validation uses the same method involving digital signatures and the PKI certificate as defined in the DOCSIS secure software download process.

If a Principal CCAP Core initiates a Remote PHY software upgrade during operational configuration or at any time during active operation, then the following events occur:

- The GCP session to the Principal CCAP Core initiating the update is terminated;
- Any L2TPv3 connections to the Principal CCAP Core initiating the update are terminated;
- Any active GCP and L2TPv3 connections to other CCAP Cores are terminated;
- The software upgrade is performed;
- The RPD reboots.

If a network management entity initiates the software upgrade, then the following events occur:

- Any active GCP and L2TPv3 connections to all CCAP Cores are terminated;
- The software upgrade is performed;
- The RPD reboots.

7.4 Software Code Upgrade Requirements

The following sections define the requirements of the RPD software code upgrade verification process. All RPD code upgrades are prepared and verified as described. All RPDs MUST verify code upgrades according to this specification. The new PKI used for issuing CVCs consists of three types of certificates: a Root CA, a CVC CA, and the CVC. CableLabs manages the new PKI and the certificates issued from its CAs (CableLabs Root CA and CableLabs CVC CA); see [SECv3.1] for certificate profile and extension definitions. The RPD MUST process CVC extensions as defined by [RFC 5280].

NOTE: The CableLabs Root CA is used to issue both RPD Device Certificates and CVC Certificates. RPDs do not support the code upgrade requirements that use the legacy PKI defined in DOCSIS 3.0.

7.4.1 Code File Processing Requirements

The code file format is defined in the [SECv3.1].

The RPD MUST reject the DOCSIS [PKCS#7] code file if the `signedData` field does not match the DER-encoded structure represented in [SECv3.1].

The RPD MUST be able to verify DOCSIS code file signatures that are signed using key modulus lengths of 1024, 1536, and 2048 bits. The public exponent is F_4 (65537 decimal).

The RPD MUST reject the CVC if it does not match the DER-encoded structure represented in [SECv3.1].

The RPD MUST NOT install the upgraded code image unless the code image has been verified as being compatible with the RPD.

If the code download and installation is successful, then the RPD MUST replace its currently stored Root CA Certificate with the Root CA Certificate in the `SignedContent` field, if one was present.

If the code download and installation is successful, then the RPD MUST replace its currently stored Device CA Certificate with the Device CA Certificate received in the `SignedContent` field, if any were present.

7.4.2 Code File Access Controls

In addition to the cryptographic controls provided by the digital signature and the certificate, special control values are included in the code file for the RPD to check before it accepts a code image as valid. The conditions placed on the values of these control parameters MUST be satisfied before the RPD attempts to validate the CVC and the CVS (see Sections 7.4.3.1 and 7.4.3.2).

7.4.2.1 Subject Organization Names

The RPD MUST recognize up to two names that it considers a trusted code-signing agent if present in the subject field of a code file CVC. These are:

- **The RPD manufacturer:** The RPD MUST verify that the manufacturer name in the manufacturer CVC subject field exactly matches the manufacturer name stored in the RPD's non-volatile memory by the manufacturer. A manufacturer CVC is always included in the code file.
- **A co-signing agent:** DOCSIS technology permits another trusted organization to co-sign code files destined for the RPD. In most cases this organization is the operator. The organization name of the co-signing agent is communicated to the RPD via a co-signer CVC via GCP when initializing the RPD's code verification process. The RPD MUST verify that the co-signer organization name in the co-signer CVC subject field exactly matches the co-signer organization name previously received in the co-signer initialization CVC, and stored by the RPD.

7.4.2.2 Time Varying Controls

In support of the code upgrade process, the RPD MUST keep two UTC time values associated with each code-signing agent. These values are known as `codeAccessStart` and `cvcAccessStart`. The RPD MUST store and maintain one pair of time values for the RPD manufacturer signing agent. If the RPD is assigned a code co-signing agent, the RPD MUST maintain a pair of time values for the code co-signing agent.

These values are used to control code file access to the RPD by individually controlling the validity of the CVS and the CVC. Stored and maintained time values in the RPD MUST have a precision of one second. Stored and maintained time values in the RPD MUST be capable of representing all times (with one second precision) between midnight, January 1 1950 and midnight January 1 2050.

The RPD MUST NOT allow the values of `codeAccessStart` and `cvcAccessStart` corresponding to the RPD's manufacturer signing agent to decrease. The RPD MUST NOT allow the value of `codeAccessStart` and `cvcAccessStart` corresponding to the co-signing agent to decrease as long as the co-signing agent does not change and the RPD maintains co-signer time-varying control values (see Section 7.4.5).

7.4.3 RPD Code Upgrade Initialization

Before the RPD can upgrade code, it should be properly initialized. The manufacturer first initializes the RPD.

7.4.3.1 Manufacturer Initialization

It is the responsibility of the manufacturer to install the initial code version in the RPD.

In support of code upgrade verification, values for the following parameters MUST be loaded into the RPD's non-volatile memory:

- RPD manufacturer organizationName;
- codeAccessStart initialization value;
- cvcAccessStart initialization value.

The RPD MUST initialize the values of `codeAccessStart` and `cvcAccessStart` to an `UTCTime` equal to the validity start time of the manufacturer's latest CVC. These values will be updated periodically under normal operation via manufacturer CVCs that are received and verified by the RPD.

7.4.3.2 Operational Initialization[13]

The method for obtaining RPD code download files is defined in Section 7.4.3. The RPD receives settings relevant to code upgrade verification from the Principal CCAP-Core via GCP. The RPD MUST NOT use these settings until after the Principal CCAP-Core has successfully initiated this process by writing "start" to the "SsdControl" object.

The GCP TLVs normally include the most up-to-date CVC applicable for the destination RPD. When the CCAP-Core initiates a code upgrade, it provides a CVC to initialize the RPD for accepting code files according to this specification. Regardless of whether a code upgrade is required, a CVC in the GCP TLVs MUST be processed by the RPD.

After the CCAP Core has successfully initiated the SSD process even if the RPD is disconnected from the Principal CCAP-Core, the SSU upgrade process is effectively enabled regardless of the settings established earlier by the Principal CCAP-Core.

GCP TLVs may contain:

- No CVCs;
- From DOCSIS 3.1 PKI:
    - A Manufacturer CVC Chain (the Manufacturer CVC and its issuing CA certificate);
    - A Co-signer CVC Chain (the Co-signer CVC and its issuing CA certificate);
    - Both Manufacturer CVC Chain and Co-signer CVC Chain

When the RPD has not received a co-signer CVC, the RPD MUST NOT accept code files that have been co-signed.

If the RPD is configured to accept code co-signed by a code-signing agent, the following parameters MUST be stored in the RPD's memory when the co-signer CVC is processed:

- Co-signing agent's `organizationName`;
- Co-signer `cvcAccessStart`;
- Co-signer `codeAccessStart`.

Unlike the manufacturer `organizationName` and time varying control values, the co-signer `organizationName` and time varying control values are not required to be stored in non-volatile memory.

[13] Revised per R-PHY-N-15.1410-2 on 1/11/16 by JB.

7.4.3.2.1 Processing the CVC Received via GCP

When a CVC is included in the GCP TLVs, the RPD MUST verify the CVC before accepting any of the code upgrade settings it contains. Upon receipt of the CVC the RPD MUST perform the following validation and procedural steps.

- If any of the following verification checks fail, the RPD MUST immediately halt the CVC verification process.
- If the GCP TLVs do not include a valid CVC, the RPD MUST NOT download upgrade code files, triggered by the GCP.

Following receipt of a CVC via GCP, and after the RPD has successfully became operational with the Principal CCAP Core, the RPD MUST:

1. Verify that the Extended Key Usage extension is present in the CVC, as specified in Appendix III of [SECv3.1].
2. Verify that the manufacturer CVC validity start time is greater than or equal to the manufacturer `cvcAccessStart` value currently held in the RPD if the CVC is a Manufacturer CVC and the subject `organizationName` is identical to the RPD's manufacturer name.
3. Reject this CVC and log an error if the CVC is a Manufacturer CVC and the subject `organizationName` is not identical to the RPD's manufacturer name.
4. Verify that the validity start time is greater than or equal to the co-signer `cvcAccessStart` value currently held in the RPD if the CVC is a Co-signer CVC and the subject `organizationName` is identical to the RPD's current code co-signing agent.
5. After the CVC has been validated, make this subject organization name become the RPD's new code co-signing agent if the CVC is a Co-signer CVC and the subject `organizationName` is not identical to the current code co-signing agent name.
6. Verify that the CVC and any CVC CA Certificate signatures chain up to the Root CA Certificate of the new PKI held by the RPD.
7. Verify that the validity periods for the CVC and the issuing CA certificate have not expired.
8. Update the RPD's current value of `cvcAccessStart` corresponding to the CVC's subject `organizationName` (i.e., manufacturer or code co-signing agent) with the validity start time value from the validated CVC. If the validity start time value is greater than the RPD's current value of `codeAccessStart`, update the RPD's `codeAccessStart` value with the validity start time value.

7.4.4 Code Signing Guidelines

Manufacturer and operator code signing guidelines are provided in Appendix III of [SECv3.1].

7.4.5 Code Verification Requirements

The RPD MUST NOT install upgraded code unless the code has been verified.

7.4.5.1 RPD Code Verification Steps[14]

When downloading code, the RPD MUST perform the verification checks presented in this section. If any of the verification checks fail, or if any section of the code file is rejected due to invalid formatting, the RPD MUST immediately halt the download process and log the error if applicable, remove all remnants of the process to that step, and continue to operate with its existing code. The verification checks can be made in any order.

1. The RPD MUST verify that:
    - The value of signingTime is equal to or greater than the manufacturer codeAccessStart value currently held in the RPD;

[14] Revised per R-PHY-N-16.1575-1 on 9/8/16 by JB.

- The value of signingTime is equal to or greater than the manufacturer CVC validity start time;
- The value of signingTime is less than or equal to the manufacturer CVC validity end time.

2. The RPD MUST verify that:
   - The manufacturer CVC subject `organizationName` is identical to the manufacturer name currently stored in the RPD's memory;
   - The manufacturer CVC validity start time is equal to or greater than the manufacturer `cvcAccessStart` value currently held in the RPD;
   - The Extended Key Usage extension in the Manufacturer CVC meets the requirements of Appendix III of [SECv3.1];
3. The RPD MUST verify that the Mfr CVC chains up to the Root CA held by the RPD.
4. The RPD MUST verify that the validity periods for the CVC and the issuing CA certificate have not expired.
5. The RPD MUST verify the manufacturer code file signature. If the signature does not verify, the RPD MUST reject all components of the code file (including the code image), and any values derived from the verification process should be immediately discarded.
6. If the manufacturer signature verifies and a co-signing agent signature is required:
   a) The RPD MUST verify that:
      (1) The co-signer signature information is included in the code file;
      (2) The value of `signingTime` is equal to or greater than the corresponding `codeAccessStart` value currently held in the RPD;
      (3) The value of `signingTime` is equal to or greater than the corresponding CVC validity start time;
      (4) The value of `signingTime` is less than or equal to the corresponding CVC validity end time.

   b) The RPD MUST verify that:
      (1) The co-signer CVC subject `organizationName` is identical to the co-signer organization name currently stored in the RPD's memory ;
      (2) The co-signer CVC validity start time is equal to or greater than the `cvcAccessStart` value currently held in the RPD for the corresponding subject `organizationName`;
      (3) The Extended Key Usage extension in the Co-signer CVC meets the requirements of Appendix III of [SECv3.1].
   c) The RPD MUST verify that the Co-Signing CVC Certificate chains up to the Root CA held by the RPD.
   d) The RPD MUST verify that the validity periods for the CVC and the issuing CA certificate have not expired.
   e) The RPD MUST verify the co-signer code file signature. If the signature does not verify, the RPD MUST reject all components of the code file (including the code image), and any values derived from the verification process should be immediately discarded.
7. Once the manufacturer, and optionally the co-signer, signature has been verified, the code image can be trusted and installation may proceed. Before installing the code image, all other components of the code file and any values derived from the verification process except the [PKCS#7] `signingTime` values and the CVC validity start values SHOULD be immediately discarded.
8. The RPD upgrades its software by installing the code file according to Section 7.3.
9. If the code installation is unsuccessful, the RPD MUST discard the [PKCS#7] `signingTime` values and CVC validity start values it just received in the code file. The procedure for handling this failure condition is specified in [MULPIv3.1].
10. Once the code installation is successful, the RPD MUST:
    a) Update the current value of manufacturer `codeAccessStart` with the [PKCS#7] `signingTime` value;

b) Update the current value of manufacturer `cvcAccessStart` with the CVC validity start value.

11. If the code installation is successful, and if the code file was co-signed, the RPD MUST:

a) Update the current value of the co-signer `codeAccessStart` with the [PKCS#7] `signingTime` value ;

b) Update the current value of the co-signer `cvcAccessStart` with the CVC validity start value.

7.4.6 DOCSIS Interoperability

Images for RPD secure software download are to be signed using certificates from the new PKI defined in the [SECv3.1] specification. Images for legacy secure software download are signed using certificates from the legacy PKI defined in [SECv3.0]are not supported by RPDs. The RPD supports secure software downloads using certificates only from the new PKI.

7.4.7 Error Codes[15]

The RPD MUST log the following error events when they occur during the code verification process. RPD event logging requirements and event message format are defined in [R-OSSI].

1. Improper code file controls

Conditions:

a) CVC subject `organizationName` for manufacturer does not match the RPD's manufacturer name.

b) CVC subject `organizationName` for code co-signing agent does not match the RPD's current code co-signing agent.

c) The manufacturer [PKCS#7] `signingTime` value is less-than the `codeAccessStart` value currently held in the RPD.

d) The manufacturer [PKCS#7] validity start time value is less-than the `cvcAccessStart` value currently held in the RPD.

e) The manufacturer CVC validity start time is less-than the `cvcAccessStart` value currently held in the RPD.

f) The manufacturer [PKCS#7] `signingTime` value is less-than the CVC validity start time.

g) Missing or improper extended key-usage extension in the manufacturer CVC.

h) The co-signer [PKCS#7] `signingTime` value is less-than the `codeAccessStart` value currently held in the RPD.

i) The co-signer [PKCS#7] validity start time value is less-than the `cvcAccessStart` value currently held in the RPD.

j) The co-signer CVC validity start time is less-than the `cvcAccessStart` value currently held in the RPD.

k) The co-signer [PKCS#7] `signingTime` value is less-than the CVC validity start time.

l) Missing or improper extended key-usage extension in the co-signer CVC.

2. Code file manufacturer CVC validation failure

Conditions:

a) The manufacturer CVC in the code file does not chain to the same root CA as the manufacturer CVC received via GCP.

3. Code file manufacturer CVS validation failure

[15] Revised per R-PHY-N-15.1410-2 on 1/11/16 by JB.

4. Code file co-signer CVC validation failure

   Conditions:

   a) The co-signer CVC in the code file does not chain to the same root CA as the co-signer CVC received via GCP.

5. Code file co-signer CVS validation failure.
6. Improper format of CVC received via GCP.

   Conditions:

   a) Missing or improper key usage attribute.

7. Validation failure of CVC received via GCP.

7.5 Security Considerations (Informative)[16]

The method(s) used to protect private keys are a critical factor in maintaining security. Users authorized to sign code, i.e., manufacturers and operators who have been issued code verification certificates (CVCs) by the DOCSIS root CA, should protect their private keys. An attacker with access to the private key of an authorized code-signing user can create, at will, code files that are potentially acceptable to a large number of RPDs.

The defense against such an attack is for the operator to revoke the certificate whose associated code-signing private key has been learned by the attacker. To revoke a certificate, the operator delivers to each affected RPD, an updated CVC with a validity start time that is newer than that of the certificate(s) being revoked. The new CVC can be delivered via any of the supported mechanisms: GCP or code file. The new CVC implicitly revokes all certificates whose validity start time is earlier than that of the new CVC.

To reduce the vulnerability to this attack, operators should regularly update the CVC in each RPD, at a frequency comparable to how often the operator would update a CRL if one were available. Regular updates help manage the time interval during which a compromised code-signing key is useful to an attacker. CVCs should also be updated if it is suspected that a code-signing key has been compromised. To update the CVC, the user needs a CVC whose validity start time is newer than the CVC in the RPD. This implies that the DOCSIS root CA regularly issues new CVCs to all authorized code-signing manufacturers and operators, to make the CVCs available for update.

When an RPD is attempting to become operational with the Principal CCAP Core for the first time or after being off-line for an extended period, it should receive a trusted CVC as soon as possible. This provides the RPD with the opportunity to receive the most up-to-date CVC available and deny access to CVCs that needed to be revoked since the RPD's last initialization. The first opportunity for the RPD to receive a trusted CVC is via GCP from the Principal CCAP Core.

To mitigate the possibility of an RPD receiving a previous code file via a replay attack, the code files include a signing-time value in the [PKCS#7] structure that can be used to indicate the time the code image was signed. When the RPD receives a code file signing-time that is later than the signing-time it last received, it will update its internal memory with this value. The RPD will not accept code files with an earlier signing-time than this internally stored value. To upgrade an RPD with a new code file without denying access to past code files, the signer may choose not to update the signing-time. In this manner, multiple code files with the same code signing-time allow an operator to freely downgrade an RPD's code image to a past version (that is, until the CVC is updated). This has a number of advantages for the operator, but these advantages should be weighed against the possibilities of a code file replay attack.

Without a reliable mechanism to revert back to a known good version of code, any code-update scheme, including the one in this specification, has the weakness that a single, successful forced update of an invalid code image may render the RPD useless, or may cause the RPD to behave in a manner harmful to the network. Such an RPD may not be repairable via a remote code update, since the invalid code image may not support the update scheme.

[16] Revised per R-PHY-N-15.1410-2 on 1/11/16 by JB.

8 X.509 CERTIFICATE PROFILE AND MANAGEMENT

R-PHY employs X.509 version 3 digital certificates for authenticating key exchanges between RPD and NAD and between RPD and CCAP Core. [X.509] is a general-purpose standard; the certificate profile, described here, further specifies the contents of the certificate's defined fields. This certificate profile also defines the hierarchy of trust for the management and validation of certificates.

Except where otherwise noted in Annex D, the certificates used comply with [RFC 5280].

8.1 Certificate Management Architecture Overview

The certificate management architecture for RPD authentication uses the DOCSIS 3.1 PKI defined by [SECv3.1]. The PKI consists of a three-level hierarchy of trust supporting three types of certificates:

- Root CA Certificate;
- Device CA Certificate;
- RPD Device Certificates.

The Root CA Certificate is used as a trust anchor for the PKI and issues the Device CA Certificate that issues the RPD Device Certificates. The PKI uses a "centralized" model where the Device CA is hosted by CableLabs or an approved 3rd party that issues RPD Device Certificates to approved manufacturers. CableLabs manages the PKI and the certificates issued from its CAs (for information about CableLabs Root CA and CableLabs Device CA, see Annex D).

The Root CA will also be used as a trust anchor for issuing and validating CA and Code Verification Certificates (CVCs) for the Secure Software Download (SSD) process specified in Section 7.

The Root CA generates and distributes to operators a Certificate Revocation List (CRL), identifying revoked manufacturer certificates. The manner in which CRLs are distributed is outside the scope of this specification. In order to reduce the burden on RPD devices that are designed to work in multiple geographic regions, an effort will be made to consolidate the DOCSIS 3.1 PKI hierarchy such that the same device certificate for DOCSIS 3.1 will also be valid for EuroDOCSIS 3.1 and other international versions of DOCSIS 3.1 and above.

8.2 RPD Certificate Storage and Management in the RPD

The RPD MUST have a factory installed RPD Device Certificate (and associated private keys) that is issued from the new PKI. The RPD uses the RPD Device Certificate when authenticating with a NAD or CCAP Core.

The RPD's non-volatile memory MUST contain a Root CA certificate for SSD image verification.

The RPD MAY be capable of updating or replacing the Device CA Certificate via the DOCSIS code download file (see Section 7).

The RPD MUST be able to process certificate serial number values containing 20 octets or fewer. The RPD MUST accept certificates that have serial numbers that are negative or zero.

8.3 Certificate Processing and Management in the CCAP Core

IKEv2 (see [RFC 7296]) employs digital certificates to verify the binding between a device's identity (encoded in a digital certificate's subject name) and its public key. The CCAP Core does this by validating the RPD Device Certificate's certification path. This path will typically consist of three chained certificates: the RPD Device Certificate, the Device CA certificate and the Root CA certificate (see section 8.1). Validating the chain follows the "Basic Path Validation" rules defined in [RFC 5280].

The CCAP Core MUST support validating certificate chains from the DOCSIS 3.1 PKI.

[RFC 4131] requires that CCAP Cores support administrative controls that allow the operator to override certification chain validation by identifying a particular CA or RPD Device Certificate as trusted or untrusted. This section specifies the management model for the exercise of these controls, as well as the processing a CCAP Core

undertakes to assess a RPD Device Certificate's validity, and thus verify the binding between the RPD's identity and its public key.

The CCAP Core MUST be able to process certificate serial number values containing 20 octets or fewer. The CCAP Core MUST accept certificates that have serial numbers that are negative or zero.

Annex D describes the format of the subject name field for each type of DOCSIS certificate. The issuer field of a certificate exactly matches the subject field of the issuing certificate. DOCSIS 3.1 PKI certificates transmitted by an RPD have name fields that conform to the format described in Annex D. A CCAP Core MUST be capable of processing the name fields of a certificate if the name fields conform to the indicated format in Annex D. A CCAP Core MAY choose to accept a certificate that has name fields that do not conform to the indicated format in Annex D.

The CCAP Core MUST process certificate extensions as defined by [RFC 5280] (see Annex D for certificate profile and extension definitions).

8.3.1 CCAP Core Certificate Management Model

The CCAP Core holds copies of the Root CA, Device CA, and RPD Device Certificates (see Section 8.1), which it obtains in one of two ways: 1) provisioning, or 2) IKEv2 messaging. Each certificate learned by a CCAP Core MUST be assigned one of four states:

- Untrusted ,
- Trusted ,
- Chained , or
- Root.

The CCAP Core MUST support the ability to provision at least two Root CA Certificates. The CCAP Core MUST support the ability to display the entire Root Certificate(s) and/or its thumbprint to the operator.

A CCAP Core learns of Device CA certificates through either the CCAP Core's provisioning interface or through receipt and processing of the client RPDs' Authentication Information messages. Regardless of how a CCAP Core obtains its Device CA certificates, the CCAP Core MUST mark them as either Untrusted, Trusted, or Chained. If a CA Certificate is not self-signed, the CCAP Core MUST mark the certificate as Chained. The CCAP Core, however, MUST support administrative controls that allow an operator to override the Chained marking and identify a given CA certificate as Trusted or Untrusted.

If a Device CA Certificate is self-signed, the CCAP Core MUST mark the certificate as either Trusted or Untrusted, according to administratively controlled CCAP Core policy.

A CCAP Core obtains copies of RPD Device Certificates in the IKEv2 messages it receives from RPDs. RPD Device Certificates are issued by a Device CA. Thus, the CCAP Core MUST mark RPD Device Certificates as Chained unless overridden by CCAP Core administrative control and configured as Trusted or Untrusted.

8.3.2 Certificate Validation

The CCAP Core validates the certification paths of CA and RPD Device Certificates using Basic Path Validation rules defined in [RFC 5280] and the criteria below.

The CCAP Core MUST label CA and RPD Certificates as Valid or Invalid if their certification paths are valid or invalid respectively. Trusted certificates, provisioned in the CCAP Core, MUST be Valid; this is true even if the current time does not fall within the Trusted certificate's validity period. Untrusted certificates, provisioned in the CCAP Core, MUST be Invalid.

The CCAP Core MUST mark a chained certificate as Valid only if:

1. The certificate chains to a Root CA, Trusted, or Valid certificate that has not been revoked as defined by the Basic Path Validation section in [RFC 5280]; and
2. The current time falls within the validity period of each Chained or Root certificate within the certificate chain; and

3. The certificate is not identified as revoked (see Section 8.4); and
4. In the case of a RPD Device Certificate, the RPD MAC address encoded in its `tbsCertificate`.subject field and RSA public key encoded in its `tbsCertificate.subjectPublicKeyInfo` field match the RPD MAC address and RSA public key encoded in the IKEv2 messaging; and
5. In the case of an RPD Device Certificate, if the KeyUsage extension is present, the digitalSignature and/or keyAgreement bits are turned on, the keyEncipherment bit is turned on, and the keyCertSign and cRLSign bits are off. In the case of a Device CA Certificate, if the KeyUsage extension is present, the keyCertSign bit is turned on.

Whether criterion 2 above is ignored MUST be subject to CCAP Core administrative control.

If validity period checking is enabled and the time of day has not been acquired by the CCAP Core, a (non-permanent) authorization reject message MUST be returned by the CCAP Core in response to an authorization request.

The CCAP Core MUST NOT invalidate certificates that have non-specified critical extensions (contrary to [RFC 5280]) as long as the certificates satisfy the validity criteria above.

8.4 Certificate Revocation

Providing a mechanism for certificate revocation is a normal part of PKI management. When a certificate is issued, it is expected to be in use for its entire validity period. However, various circumstances may cause a certificate to become invalid prior to the expiration of the validity period. Such circumstances include change of name, change of association between subject and CA, and compromise or suspected compromise of the corresponding private key. Under such circumstances, the CA needs to revoke the certificate.

Two methods of supporting certificate revocation are defined in this specification: 1) Certificate Revocation Lists (CRLs), and 2) Online Certificate Status Protocol (OCSP). The CCAP Core MUST support configuration of none, one, or both certificate revocation methods to be enabled at the same time.

8.4.1 Certificate Revocation Lists

[RFC 5280] defines a method for revoking certificates using [X.509] Certificate Revocation Lists (CRLs).

Figure 20 shows a framework for managing and distributing CRLs. A CRL is a digitally signed, timestamped list of certificate serial numbers revoked by a Certificate Authority (CA). When a CA identifies the compromised certificates, the CA could generate the CRLs itself, or a CA could delegate the CRL generation to a third party CRL Issuer. The CRL Repository is a system that maintains a database of revoked certificates. A description of the interface between the CA or CRL Issuer and CRL Repository is outside the scope of this specification.

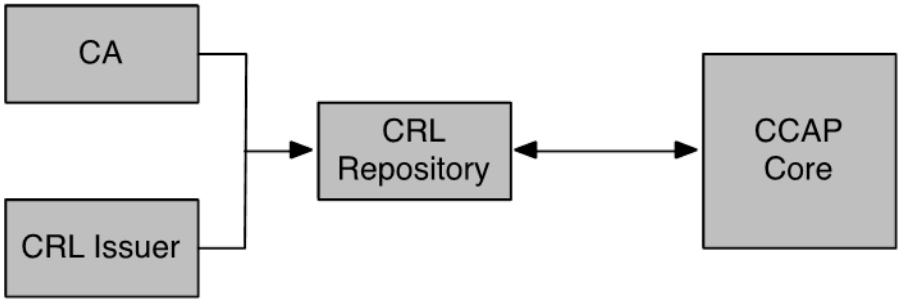

Figure 20 - CRL Framework

The CCAP Core retrieves CRL entries from the CRL Repository and uses this information to verify if a certificate received during the RPD authentication process is revoked.

8.4.1.1 CCAP Core CRL Support

The CCAP Core MUST support retrieval of CRL files formatted as defined in [RFC 5280]. CRL files may identify revoked certificates that were issued from different CAs. Therefore, the CCAP Core MUST support extensions related to indirect CRL files, as defined in [RFC 5280]. The CCAP Core MUST support HTTP as defined in [RFC 2616] for downloading CRL files.

Before using the information in a CRL file, the CCAP Core MUST verify that its digital signature chains to a trusted root CA. Trusted root CAs are administratively provisioned in the CCAP Core. If the CRL file digital signature cannot be verified, the CCAP Core MUST discard the CRL file. The CCAP Core MUST validate if a CA certificate or RPD Device Certificate is revoked during the certificate validation process specified in Section 8.3.2.

If the CRL contains the nextUpdate value, the CCAP Core MUST refresh the CRL after the specified time has passed. If the CCAP Core fails to retrieve the new CRL, it MUST log an event (see [CCAP-OSSIv3.1]) and continue to use its current CRL. If the CCAP Core fails to retrieve the new CRL it should attempt to retry retrieval of the CRL file on a periodic basis. If the CRL does not contain the nextUpdate value, the CCAP Core MUST refresh the CRL according to the configured value as defined in [CCAP-OSSIv3.1].

When the CCAP Core is configured to use a CRL it MUST attempt to retrieve the CRL file each time it starts up. During CCAP Core startup it is possible that some RPDs may perform IKEv2 authorization before the CRL file has been retrieved. When the CCAP Core is configured to use a CRL and an RPD's device certificate chain is validated during CCAP Core startup before the CRL file is retrieved, the CCAP Core MUST log an event for that RPD [CCAP-OSSIv3.1] and bypass CRL checking.

8.4.2 Online Certificate Status Protocol

[RFC 6960] defines an Online Certificate Status Protocol (OCSP) for querying the status of a digital certificate. The CCAP Core sends a certificate status request to an OCSP responder when it receives a CA certificate or an RPD Device Certificate (see Figure 21). The OCSP responder sends a status response indicating that the certificate is either "good," "revoked," or "unknown." The OCSP responder checks only the revocation status of a certificate; it does not verify the validity of the certificate itself. The CCAP Core uses the result from the OCSP responder during the certificate validation process specified in Section 8.3.2.

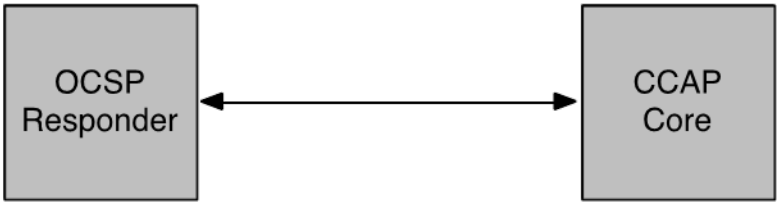

Figure 21 - OCSP Framework

The CCAP Core MUST be capable of acting as an OCSP client as defined in [RFC 6960]. The CCAP Core SHOULD cache the OCSP response status for a certificate if the nextUpdate value is present in the OCSP response. If the CCAP Core caches the OCSP response status for a given certificate, it MUST retrieve the revocation status from the cache. Once the nextUpdate time for that certificate has passed, the CCAP Core MUST continue using the revocation status value from the cache until an update is retrieved from the OCSP Responder. If the CCAP Core is unable to retrieve the OCSP status for an uncached certificate or if the retrieved status is "unknown," the CCAP Core MUST log an event [CCAP-OSSIv3.1] and assume the certificate status to be "good."

If the nextUpdate value is not present in the OCSP response, the CCAP Core MUST NOT cache the OCSP response status for a certificate. If the CCAP Core is configured with OCSP Responder information, it MUST send an OCSP request when a CA certificate or RPD Device Certificate is obtained using the Authentication Information message, or Authentication Request message respectively, unless there is a valid certificate status in the cache.

When the CCAP Core is attempting to communicate with the OCSP Responder, the exchange should not significantly delay the RPD provisioning process. If no response is received, the CCAP Core MUST proceed using the currently cached revocation status. For uncached certificate states, the CCAP Core MUST proceed as if a response with the status "good" has been received.

The CCAP Core MUST support OCSP over HTTP as described in [RFC 6960]. The CCAP Core MAY generate a signature in the OCSP request. The CCAP Core MUST bypass validation of the signature in an OCSP response based on the configured value as defined in [CCAP-OSSIv3.1].

9 PHYSICAL PROTECTION OF KEYS IN THE RPD

RPDs MUST store and maintain the RPD Device Certificate RSA private/public key pairs. The RPD MUST store the RPD Device Certificate private keys in a manner that deters unauthorized disclosure and modification. Also, RPDs SHOULD prevent debugger tools from reading the RPD Device Certificate private key in production devices by restricting or blocking physical access to memory containing this key.

The RPD MUST meet [FIPS 140-2] security requirements for all instances of private and public permanent key storage.

The RPD MUST meet [FIPS 140-2] Security Level 1. FIPS 140-2 Security Level 1 requires minimal physical protection through the use of production-grade enclosures. The reader should refer to the cited document for the formal requirements; however, below is a summary of those requirements.

Under the [FIPS 140-2] classification of "physical embodiments" of cryptographic modules, external RPDs are "multiple-chip stand-alone cryptographic modules. FIPS 140-2 specifies the following Security level 1 requirements for multiple-chip stand-alone modules:

- The chips are to be of production-grade quality, which shall include standard passivation techniques (i.e., a sealing coat over the chip circuitry to protect it against environmental or other physical damage);
- The circuitry within the module is to be implemented as a production grade multiple-chip embodiment (i.e., a printed circuit board, a ceramic substrate, etc.);
- The module is to be entirely contained within a metal or hard plastic production-grade enclosure, which may include doors or removable covers.

10 SYSTEM OPERATION (NORMATIVE)

Once the system is operational, there is very little that happens with the MHAv2 protocols. Most of the operational features are managed within the DOCSIS protocol that is run transparently over MHAv2.

This section explains some variations to the MHAv2 operational state. One of those variations is the location of the upstream scheduler.

10.1 DOCSIS Upstream Scheduling

The RPD is intended to be a simple and lightweight extension of the CCAP. MHAv2 permits the upstream scheduler to be located either centrally or remotely. Note that the [R-UEPI] protocol provides sufficient quality of service mechanisms that the upstream scheduler can be run centrally.

The advantages of running a centralized upstream scheduler are:

- Similar CMTS software model to an Integrated CMTS.
- Similar operational model to an Integrated CMTS.
- Scheduler software is from the same vendor as the CMTS software.
- Fewer interoperability problems between different vendors of CCAP Core and the RPD.
- Access to Debug mode if there are problems with the remote scheduler.
- Scalable resources for the scheduler if more CPU power is needed.

The advantages of running a distributed upstream scheduler are:

- Shorter round-trip delay from request to grant that may impact some aspects of performance.

For plant distances of 100 miles or less, the Remote PHY and I-CMTS systems have nearly identical performance as the I-CMTS, since the I-CMTS is a centralized scheduler system (because the PHY is also centralized). With the PHY removed from the CMTS Core, the CMTS Core can be located at distances much greater than the original 100 mile limit for DOCSIS. In these cases, the REQ-GNT turnaround time could be extended by several additional milliseconds. However, the DOCSIS scheduler is a pipelined system. If the time between grants increases, then the number of bytes per grant will increase to compensate.

The R-PHY system defaults to a centralized scheduler because the differences in performance are negligible and the benefits are measureable. Support for a distributed scheduler is not included at this time.

10.1.1 Centralized Scheduling Requirements

The requirements regarding centralized scheduling are:

- The RPD MUST support operation with a centralized scheduler.
- The RPD MAY support operation with a distributed scheduler.

10.2 Daisy-chaining of the Backhaul Ethernet Port

The RPD may be located in a node enclosure with other entities that aggregate to the same backhaul link to the CIN. Two distinct forms of aggregation are supported:

1. All RPDs connect to an Ethernet switch or hub which then connects to the CIN.
2. Each RPD is daisy-chained with the next RPD, and the last RPD connects to the CIN. In the case of daisy-chaining, it is as if each RPD has a three-port Ethernet switch associated with it that lets traffic either pass through, or to be injected/removed by the device.

10.2.1 Backhaul Daisy-chaining Requirements

The requirements regarding backhaul daisy-chaining are:

- Each RPD that is to be individually authenticated MUST have its own MAC address and its own IP address assignment.
- When operating in a daisy-chained topology, the RPD MUST support the authentication requirements defined in Section 6.3.

10.3 Networking Considerations

It is important to distinguish between the terms "PHB-ID" and "DSCP" as used in the MHAv2 specifications:

- A "PHB-ID" is a 6-bit value appearing in an L2TPv3 Attribute Value Pair (AVP)
- A "DSCP" is a 6-bit value appearing in an IP packet header

All L2TPv3 packets in [R-DEPI] are in a control session, a PSP session, or a non-PSP session. All PSP sessions contain both downstream and upstream data "flows". PHB-IDs apply to flows of PSP data sessions. DSCPs apply to the IP packets that encapsulate all L2TPv3 packets, i.e., DSCPs apply to control sessions, PSP sessions, and non-PSP sessions.

For a downstream PSP flow, the CCAP Core assigns via L2TPv3 AVPs the PHB-ID for each downstream PSP flow. The assigned downstream flow PHB-ID selects the scheduling behavior for that flow *only on the RPD*, i.e., for the scheduling of multiple downstream PSP flows on the single hop from CIN to RF network. At a minimum, an RPD SHOULD provide highest-priority strict priority service to PSP flows assigned to the Expedited Forwarding(46) PHB-ID. Support for more complex scheduling disciplines, e.g., multiple strict priorities or weighted fair queueing, is for further study.

The RPD advertises via GCP to the CCAP Core what PHB-IDs it supports for downstream PSP flow scheduling. An RPD MUST support at least the Expedited Forwarding (46) and BestEffort(0) PHB-IDs. The use of PHB-IDs other than ExpeditedForwarding(46) and BestEffort(0) is for further study.

For a downstream PSP flow, The CCAP Core selects the DSCP to send in the IP header of the L2TPv3 data session packets for the flow. The DSCP selects the per-hop behavior *on each CIN router* between the CCAP Core and RPD. The 6-bit DSCP of the IP headers of downstream L2TPv3 data packets on a PSP flow may or may not equal the 6-bit PHB-ID assigned to the flow on the RPD itself. For example, the CCAP Core may use more than two different DSCP values when the CIN supports them. The RPD ignores the DSCP of a downstream IP packet and uses only the flow ID in the inner PSP sub-layer to select the queue with which it schedules downstream data for the flow.

For an upstream PSP flow, the CCAP Core assigns via L2TPv3 AVPs the PHB-ID for each upstream PSP flow. For the upstream case, the PHB-ID corresponds to a "recommended DSCP value" as described in [RFC 3140]. The RPD sets the DSCP in the IP headers of all upstream L2TPv3 data packets for a PSP flow to the PHB-ID value assigned to that flow. The PHB-ID assigned to an upstream PSP flow does not identify any per-hop behaviour in the RPD itself.

10.3.1 Per Hop Behavior

The IETF has defined a number of Per Hop Behaviors (PHBs) to be used for offering network-based QoS. DEPI supports use of the 6-bit Expedited Forwarding (EF) PHB as described in [RFC 3246], Assured Forwarding (AF) PHBs as described in [RFC 2597], and best effort forwarding as described in [RFC 2597]. DEPI negotiates six-bit Per-Hop Behavior Identifiers (PHBIDs) between the CCAP Core and the RPD.

The RPD advertises the PHB-IDs it supports for its downstream PSP packet scheduler. The RPD MUST support Expedited Forwarding(46) and BestEffort(0) PHB-IDs. The RPD SHOULD provide highest strict priority scheduling service to PSP flows assigned to the Expedited Forwarding(46) PHB-ID. The CCAP Core SHOULD support assigning the Expedited Forwarding(46) PHB-ID to a separate PSP flow for MAPs+UCDs with Best Effort (0) for all other traffic.

For upstream flows, the RPD MUST support signaling of an arbitrary 6-bit PHB-ID as the transmitted 6-bit DSCP value.

NOTE: Table 2 lists the PHBs explicitly supported by the DEPI specification. This specification does not prohibit support for other PHBs not defined in Table 2.

Table 2 - PHBs and Recommended DSCP Values

PHB	PHB ID(s) and Recommended DSCP Value(s)
EF	46
AF (multiple levels)	10, 12, 14, 18, 20, 22, 26, 28, 30, 34, 36, 38
Best effort	0

The DEPI interface supports multiple traffic types including DOCSIS MAC and DOCSIS data traffic. Within both traffic types, there may be different levels of priority. For PSP operation, the CCAP Core SHOULD provide a mechanism to map traffic of different priorities to DEPI flows with different PHB values. The CCAP Core SHOULD NOT use the same PHB across multiple DEPI flows within a session.

The CIN should provide the appropriate Per Hop Behavior for the differentiated traffic types. The level of granularity provided for differentiated traffic is determined by the network operator, but at a minimum, it is expected that DOCSIS MAP messages and VoIP data traffic are prioritized higher than best effort data traffic.

The RPD uses the PHB signaled in the establishment of the DEPI flow when scheduling multiple DEPI PSP flows onto one QAM channel as described in [R-DEPI].

10.3.2 DiffServ Code Point Usage

An operator sets up CIN network elements to support a particular set of DSCPs. The selected DSCPs should select appropriate per-hop behavior at each network element for differentiated traffic types.

For L2TPv3 data sessions, packets in the same direction have the same DSCP; packets in different directions may have different DSCPs. For PSP L2TPv3 sessions, each PSP "flow" in the session and in a particular direction may have a different DSCP value. Different PSP flows in the same PSP session may have the same DSCP.

The CCAP Core is responsible for selecting the DSCP values of all L2TPv3 control and data session packets, including the DSCP sent by the RPD.

The CCAP Core and RPD MUST:

- set the same DSCP on all L2TPv3 control packets;
- set the same DSCP on all L2TPv3 data packets of the same non-PSP session in a direction;
- set the same DSCP on all L2TPv3 data packets of the same PSP flow in a direction.

DOCSIS frames encapsulated in L2TPv3 packets may contain IP packets which also have a DSCP assigned. The RPD is not required to schedule packets based upon the original DSCP contained within the DOCSIS frame.

10.3.3 Packet Sequencing

For a stream of packets transmitted on a DEPI flow, the packet sequence number is incremented by one for each packet sent, as described in [R-DEPI].

If the RPD detects a discontinuity in the packet sequence numbers indicating that one or more packets were dropped or delayed, an error is logged and the RPD SHOULD transfer the current packet to the QAM channel without waiting for the missing packets. If the RPD detects a discontinuity in the packet sequence numbers indicating that one or more packets have arrived late, those packets SHOULD be discarded.

The RPD MUST NOT forward packets that were skipped due to a discontinuity in the sequence numbers. Storing and re-ordering of packets so that they can be delivered to the QAM channel in the correct sequence is not prohibited by these requirements and the RPD MAY perform such re-ordering as long as the latency requirements of Section 5.6 are met.

10.3.4 Network MTU

The network between the CCAP Core and the RPD has a certain Maximum Transmission Unit (MTU). If a maximum size DOCSIS frame were to be tunneled from the CCAP Core to the RPD without fragmentation, the size

of the resulting packet could be greater than the CIN can handle. Both the D-MPT and PSP modes avoid this issue by offering streaming and fragmentation. As such, IP fragmentation is not required. IP fragmentation is also undesirable because the RPD may forward packets based upon the destination UDP port, and the UDP port is only available in the first IP fragment.

Determining the MTU to use for the L2TPv3 tunnel between the CCAP Core and the RPD is a two-step process:

1. Choose the payload size.
2. Determine the MTU of the path between the CCAP Core and the RPD.

The first step is done as part of L2TPv3 session establishment (see [R-DEPI]) using the DEPI MTU AVPs. When the CCAP Core sends the session ICRQ message it MUST supply the DEPI Local MTU AVP with a payload size that is the lesser of its receive capabilities and the receive capabilities defined by its lower layer. The receive capabilities of the CCAP Core are defined by its internal constraints, and any configured maximums. The receive capabilities defined by its lower layer are calculated based on referencing the payload size constraints of the interface below which this tunnel is being created, as defined in Annex A.1.

The CCAP Core MUST support an MTU size of at least 2000 bytes, as calculated in Annex A.1. The RPD MUST send L2TPv3 frames with a payload size less than or equal to this maximum. If the RPD cannot meet this criterion then it MUST fail session creation by generating a CDN message. The RPD needs to consider the same criteria in calculating its MTU.

The RPD MUST support an MTU size of at least 2000 bytes, as calculated in Annex A.1. The RPD MUST insert the DEPI Remote MTU AVP in the ICRP message with its MTU size. The CCAP Core MUST send L2TPv3 frames with a payload size less than or equal to this maximum. If the CCAP Core cannot meet this criterion then it MUST fail session creation by generating a CDN message.

The second step is to determine the MTU of the path between the CCAP Core and the RPD. The CCAP Core MUST provide a mechanism to prevent sending packets larger than the network MTU. This SHOULD be done using Path MTU Discovery, as described in [RFC 1191]. Annex A.3 gives a brief overview of the Path MTU discovery protocol.

Alternatively, this MAY be done via a static configuration option. Both the CCAP Core and the RPD MUST have a way to statically configure an MTU for each L2TPv3 session. To avoid IP fragmentation, the CCAP Core and the RPD MUST set the Don't Fragment bit (DF) in the IPv4 header for all transmissions into the L2TPv3 pseudowire.

11 MULTIPLE CCAP CORE OPERATION

11.1 Introduction

The MHAv2 architecture permits RPDs to be managed by more than one CCAP Core. An RPD is controlled by exactly one "principal" CCAP Core and zero or more "auxiliary" CCAP Core(s). An "auxiliary" core manages a subset of RPD resources, e.g., particular channels or RF ports. Each auxiliary CCAP Core establishes its own GCP session and L2TPv3 control sessions with the RPD. The specification term "CCAP Core" can refer to either the principal core or an auxiliary core.

Potential auxiliary CCAP Cores include but are not limited to the following:

- A "Broadcast EQAM" CCAP Core that controls only downstream video broadcast channels;
- A "Narrowcast EQAM" CCAP Core that controls only downstream video narrowcast channels;
- A "Forward OOB" CCAP Core that controls and sources NDF, typically broadcast to multiple RPD ports;
- A "Reverse OOB" CCAP Core that controls and receives NDR channels, always received one per RPD port.
- A CMTS CCAP that controls the downstream and upstream channels of a separate MAC domain;

A CMTS Core is programmed or configured in a vendor-specific manner to operate as an auxiliary core.

11.2 RPD Startup with Multiple Cores

An RPD MUST implement an *ActivePrincipalCore* object into which the principal CCAP Core controlling the RPD writes its IP address.

Attribute Name	Type	Access	Type Constraints	Units	Default
ActivePrincipalCore	IpAddress	RW			

An RPD MUST implement an *ActiveAuxCoreTable* object into which auxiliary CCAP Cores connected to the RPD write their IP addresses.

Attribute Name	Type	Access	Type Constraints	Units	Default
ActiveAuxCoreTable		RW			
ActiveAuxCoreIp	IpAddress	RW	key		

An RPD MUST implement a *ConfiguredCoreTable* object that contains the list of principal and auxiliary cores to which the RPD attempts to attach. This table is originally populated by the RPD itself based on DHCP, but may be modified by the principal core.

Attribute Name	Type	Access	Type Constraints	Units	Default
ConfiguredCoreTable		RW			
ConfiguredCoreIp	IpAddress	RW	key		

A resetting RPD MUST clear its *ConfiguredCoreTable, ActivePrincipalCore* and *ActiveAuxiliaryCoreTable* objects. This requirement applies to both cold-reset and warm-reset.

An RPD MUST accept up to six *CCAP Core-IP-Address* options (a new [CANN] 42.x option) in its DHCP response. A starting RPD initially populates its *ConfiguredCoreTable* with the *CCAP Core-IP-Address* list learned from DHCP.

After completing initial DHCP, an RPD MUST attempt to establish EAP-TLS authentication and a GCP TCP session with each of the *ConfiguredCoreTable* IP addresses until a Principal Core identifies itself by writing to the *ActivePrincipalCore* RPD object.

After a GCP TCP session is established, the Principal CCAP Core MUST write its IP address into the *ActivePrincipalCore* object of the RPD before attempting any other GCP write operations.

After a GCP TCP session is established, an auxiliary CCAP Core MUST add its IP address to the RPD's *ActiveAuxiliaryCores* table. The Principal Core MAY add IP addresses to the *ActiveAuxiliaryCores* table.

The RPD MUST attempt to maintain an IPsec-authorized GCP session to each core IP address in its *ActivePrincipalCore* object and *ActiveAuxiliaryCores* table.

11.3 Downstream Channel Constraint Table

Downstream QAM modulators are often implemented with hardware "channel blocks" that constrain consecutively identified channels to have the same or related physical attribute. A Principal CCAP Core attempting to dynamically determine resource sets of downstream channels is made aware of those constraints by a read-only object table on the RPD.

An RPD MUST implement a read-only *DownChannelConstraintTable* to identify constraints imposed by its hardware on the configuration of physical parameters of blocks of downstream channels.

The set of described constraints are implied as present on all downstream RF ports of the RPD.

Attribute Name	Type	Access	Type Constraints	Units	Default
DownChannelConstraintTable	NA				
Index	unsignedInt		key		
DownChanIndexStart	unsignedInt		0..159		
DownChanIndexEnd	unsignedInt		0..159, > DownChanIndexStart		
LockParameters	LockParamBits				

The constraints are defined as a `LockParameters` bitmask:

```
LockParameters EnumBits {
   frequency(0),
   bandwidth(1),
   power(2),
   modulation(3),
   interleaver(4),
   j83Annex(5),
   symbolRate(6)
   mute(7)
}
```

The `LockParameters` field is a bitmask from which constraints apply to the range of channels defined by `DownChanIndexStart` through `DownChanIndexEnd`, inclusive. Note that different *DownChannelConstraintTable* objects may describe different `LockParameter` values on overlapping or partially overlapping channel ranges of other `DownChannelContraintTable` objects:

"frequency(0)" means the channels are constrained to have consecutive frequencies;

"bandwidth(1)" means the channels are constrained to have the same channel width;

"power(2)" means the channels are constrained to have the same power adjustment;

"modulation(3)" means the channels are constrained to have the same modulation;

"interleaver(4)" means the channels are constrained to have the same interleave value;

"j83Annex(5)" means the channels are constrained to have the same j83Annex definition;

"symbolRate(6)" means the channels are constrained to have the same symbol rate;

"mute(7)" means the channels are constrained to be muted or unmuted together.

11.4 Resource Sets and Auxiliary Resource Assignment

An RPD MUST implement the `ResourceSet` table , which identifies which auxiliary cores may control which RPD object.

Attribute Name	Type	Access	Type Constraints	Units	Default
ResourceSet					
ResourceSetIndex	Integer	RW	key		
DsRfPortStart	Integer	RW	-1 if unused		
DsRfPortEnd	Integer	RW	-1 if unused		
DsChanIndexStart	Integer	RW	-1 if unused		
DsChanIndexEnd	Integer	RW	-1 if unused		
UsRfPortStart	Integer	RW	-1 if unused		
UsRfPortEnd	Integer	RW	-1 if unused		
UsChanIndexStart	Integer	RW	-1 if unused		
UsChanIndexEnd	Integer	RW	-1 if unused		

A resource set consists of a range of channels from start to end (inclusive) on particular RF ports from start to end (inclusive). The RPD MUST enforce that no two entries in the `ResourceSet` table overlap, i.e., include the same channel on the same RF port.

A Principal CCAP Core MUST implement R-OSSI tables that permit CCAP Core configuration of the `ResourceSet` table objects to be written to an RPD.

An RPD MUST implement the `AuxResourceAssignment` table, which identifies which auxiliary core is authorized to manage which resource set. The RPD MUST implement sufficient entries in `AuxResourceAssignment` to assign each supported auxiliary core to one resource set.

Attribute Name	Type	Access	Type Constraints	Units	Default
AuxResourceAssignment	List				
ResourceSetIndex	key	RW	See text.		
AuxCoreIpAddress	key	RW			

The RPD MUST permit more than one `AuxCoreIpAddress` to be configured to the same `ResourceSetIndex`.

An RPD MUST implement the `PermitAuxSelfConfiguration` object to control whether auxiliary cores are permitted to configure their own resource sets.

Attribute Name	Type	Access	Type Constraints	Units	Default
PermitAuxSelfConfiguration	Boolean	RW	Writeable by Principal Core only		false

An RPD MUST enforce the policy that only the Principal Core can write to `PermitAuxSelfConfiguration`.

The Principal Core is by default solely responsible for writing the `ResourceSet` and `AuxResourceAssignment` tables. When `PermitAuxSelfConfiguration` is 'false' (the default), the RPD MUST enforce that only the Principal Core may write to the `ResourceSetTable` and `AuxResourceAssignmentTable`. When `PermitAuxSelfConfiguration` is 'true' (as changed by the Principal Core) the RPD MUST permit any auxiliary core to write to `ResourceSetTable` and to assign those entries to its own IP address by writing to the `AuxResourceAssignment` table. Even when `PermitAuxSelfConfiguration` is 'true', the RPD MUST enforce the policy that an auxiliary core writes only its own IP address into the `AuxResourceAssignment` table.

11.5 RPD Reads and Writes

The RPD MUST support concurrent reads of any objects by any connected CCAP Core, whether a Principal Core or an Auxiliary Core.

The RPD MUST reject an attempt by an Auxiliary Core to write to any object not specifically authorized to it by the `AuxResourceAssignment` table. The RPD MUST reject an attempt by the Principal Core to write to any object assigned to an Auxiliary Core in the `AuxResourceAssignment` table.

12 REMOTE PHY PNM FUNCTIONS[17]

The majority of DOCSIS 3.1 PNM functions are already supported "in-band" by the R-PHY data plane protocols. There are however two PNM functions, which are distributed in nature and require additional control plane instrumentation. These are the Downstream Symbol Capture and the Upstream Histogram. The following sections describe the decomposition of functionality between the RPD and the CCAP Core and the protocol necessary to implement Downstream Symbol Capture and Upstream Histogram functions in the R-PHY architecture.

12.1 Downstream Symbol Capture

The DOCSIS 3.1 PNM Downstream Symbol Capture provides the equivalent functionality of a network analyzer for analyzing the response of a cable plant on the downstream. By simultaneously capturing the input to a cable plant at the CMTS and the output at the CM, one can derive the channel response in the frequency range covered by the downstream OFDM channel.

For proper comparison the CMTS and CM need to capture the same symbol in the same PLC frame. In the I-CCAP architecture, the MAC provides signaling via the PLC Trigger Message to ensure that the same symbol is captured at the CMTS and CM. The PLC Trigger Message includes fields for frame delay and symbol select, which specify respectively the number of PLC frames to delay and the symbol for which the symbol capture should be performed at both the CMTS and CM.

In the R-PHY architecture, the I-CCAP is replaced with two distinct components: the CCAP Core and the RPD. As a result, the DS symbol capture function in the CMTS is also replaced by two components: one component resides in the CCAP Core and the other in the RPD. The component in the CCAP Core interfaces with the PNM server while the component in the RPD provides the synchronization mechanism for the CM as well as capturing the cable plant input.

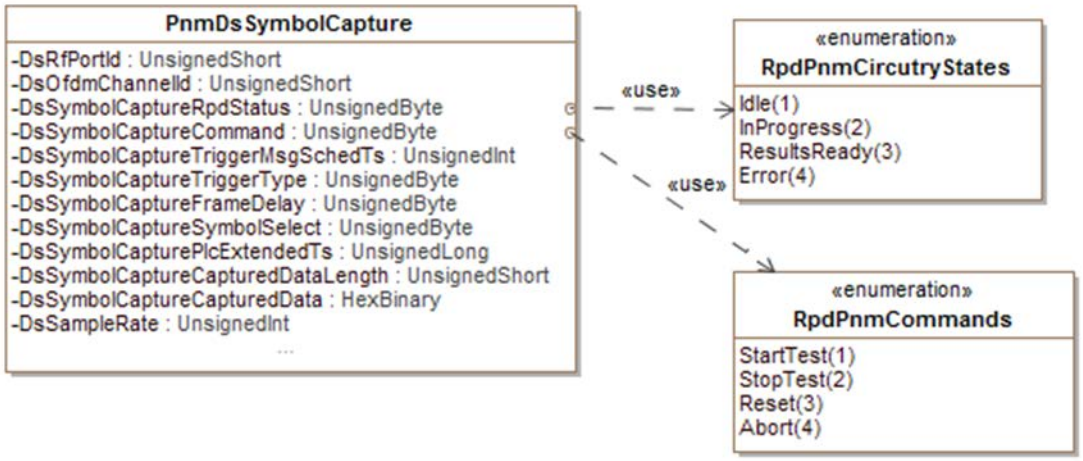

Figure 22 - GCP Objects used in DS symbol capture.

Figure 22 shows the GCP objects used to facilitate the DS symbol capture control protocol. The objects DsRfPort and DsOfdmaChannelId uniquely identify the OFDM channel in the RPD. The DsSymbolCaptureRpdStatus is a read-only object through which the CCAP Core can determine the current state of the RPD test circuitry. The protocol defines four states of the RPD's downstream symbol capture circuity: "Idle", "InProgress", "ResultsReady" and "Error". The object "DsSymbolCaptureCommand" permits the CCAP Core to instruct the RPD to start, stop, reset and abort the test.

The object DsSymbolCaptureTriggerMsgSchedTs represents the 32-bit timestamp which uniquely identifies the PLC frame in which the RPD will transmit the Trigger Message. The same value is appended to the Trigger

[17] Added all new Section 12 and subsections per R-PHY-15.1401-1 on 1/8/2016 by JB.

Message when the CCAP Core transmits it over the PLC pseudowire. The objects DsSymbolCaptureTriggerType, DsSymbolCaptureFrameDelay and DsSymbolCaptureSymbolSelect represent the parameters from the Trigger Message. These objects are communicated over GCP to permit an implementation in which the RPD does not analyze the Trigger Message sent on the PLC. The object representing the Triger Group field is purposely excluded because it is not relevant in control plane protocol between the RPD and the CCAP Core. The RPD reports the actual timestamp inserted in the PLC frame in which the Trigger Message is transmitted through object DsSymbolCapturePlcExtendedTs. The objects DsSymbolCaptureCapturedDataLength and DsSymbolCaptureCapturedData are used by the CCAP Core to read the captured data from the RPD. The DsSampleRate object conveys the sampling rate used in DS Symbol Capture.

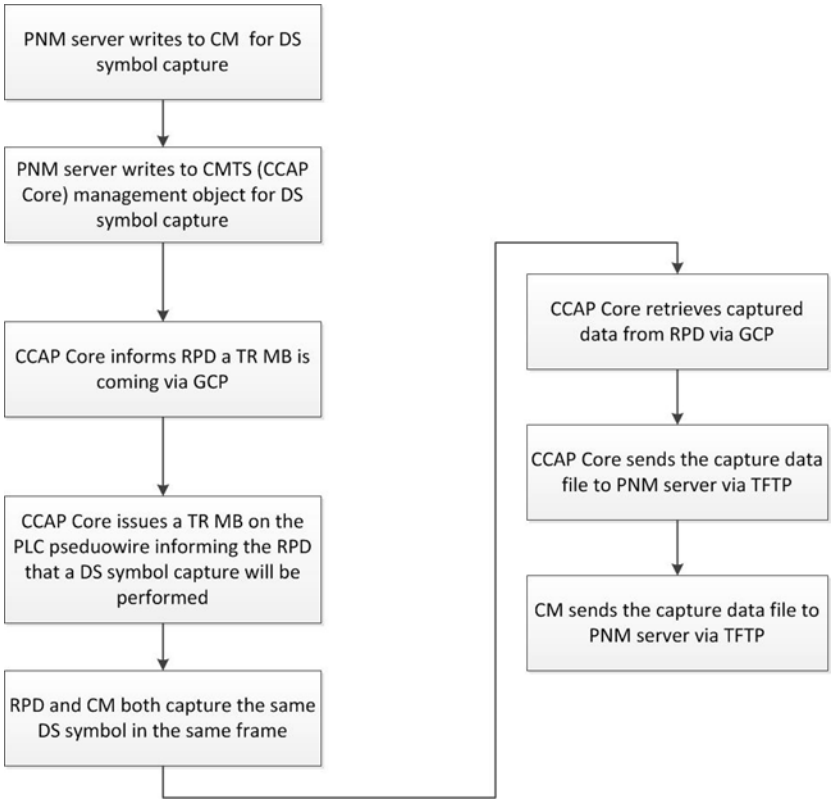

Figure 23 - DS Symbol Capture Flow in R-PHY Architecture

Figure 23 shows the flow of DS symbol capture in the R-PHY architecture. The steps for DS symbol capture in R-PHY are:

1. PNM server writes to CM MIB objects, informing the CM of a coming DS symbol capture event.

2. PNM server writes to CCAP Core MIB objects, preparing the CCAP Core for DS symbol capture.

3. CCAP Core writes “StartTest” to RPD DsSymbolCaptureCommand GCP object, informing the RPD of a coming PLC Trigger Message for a selected OFDM channel. The written objects identify the selected OFDM channel, as well as the same 32-bit timestamp, frame delay and symbol select parameters as in the PLC trigger message sent in the next step. In this step the CCAP also writes the following GCP objects:

- DsRfPort and DsOfdmaChannelId to identify the channel on which the test is performed.
- DsSymbolCaptureTs, DsSymbolCaptureTriggerType, DsSymbolCaptureFrameDelay and DsSymbolCaptureSymbolSelect to convey the parameters sent in the Triger message and the to identify the PLC frame in which the Trigger Message is sent.
- DsSymbolCaptureCommand to start the test.

The CCAP Core can write to all these objects in a single GCP message.

4. CCAP Core sends RPD a PLC Trigger Message prepended by a timestamp through a PSP session. The RPD subsequently places the Trigger Message in the PLC frame according to its prepended timestamp.

5. RPD and CM perform the DS symbol capture at the same PLC frame and symbol as directed by the PLC trigger message and GCP objects.

6. The CCAP Core retrieves the captured data from the RPD through GCP. This is accomplished in two sub-steps:

a. CCAP Core confirms availability of captured data by reading DsSymbolCaptureRpdStatus object and verifying that the status reports "ResultsReady" value.

b. The CCAP Core subsequently retrieves captured data from RPD through GCP. The CCAP Core resets the RPD DS Symbol circuitry after step 6 by writing the "Reset" value to the DsSymbolCaptureCommand object.

7. CCAP Core sends PNM server its captured data via TFTP.

8. CM sends PNM server its capture data.

Steps 1, 4 and 8 guarantee that the CM performs DS symbol capture in a similar fashion for both I-CCAP and R-PHY architectures. Steps 1, 2, 7, and 8 permit the same implementation of DS symbol capture on the PNM server side for both I-CCAP and R-PHY architectures. Step 3 enables a simplified implementation of the RPD as it does not need to decode PLC Trigger Message sent through the PLC pseudowire. If a particular implementation of an RPD is capable of decoding the Trigger Message, it may ignore Step 3 as the information carried in Step 3 is identical to Step 4. In addition, to allow RPD enough time to prepare, the CCAP Core MUST be carry out Step 3 at least 0.5 second prior to Step 4. The RPD MUST make the results available within 0.5 seconds after transmission of the Trigger Message.

The CCAP Core MAY instruct the RPD to abort the test at any time by writing "Abort" value to DsSymbolCaptureCommand object.

In order to minimize the impact on the RPD and the normal operation of a cable plant, DS symbol capture SHOULD be conducted sequentially. The CCAP Core SHOULD only initiate a DS symbol capture for an RPD after it completes or aborts the previous one. The CCAP Core MUST be capable of capturing one DS symbol per system. The RPD MUST be capable of capturing one DS symbol per device.

12.2 Upstream Histogram

The DOCSIS 3.1 PNM Upstream Histogram provides a measurement of nonlinear effects in the Upstream OFDM channel, such as amplifier compression and laser clipping. In an I-CCAP environment, the CMTS collects time domain samples at the output of the wideband receiver to calculate a histogram of 255 or 256 equally spaced bins depending on odd or even symmetry. In an R-PHY architecture, the I-CCAP is replaced with two distinct components: the CCAP Core and the RPD. As a result, the US Histogram function in the CMTS is also replaced by two components: one component resides in the CCAP Core and the other in the RPD. The component in CCAP Core interfaces with the PNM server while the component in the RPD collects the wideband receiver output and generates the actual histogram.

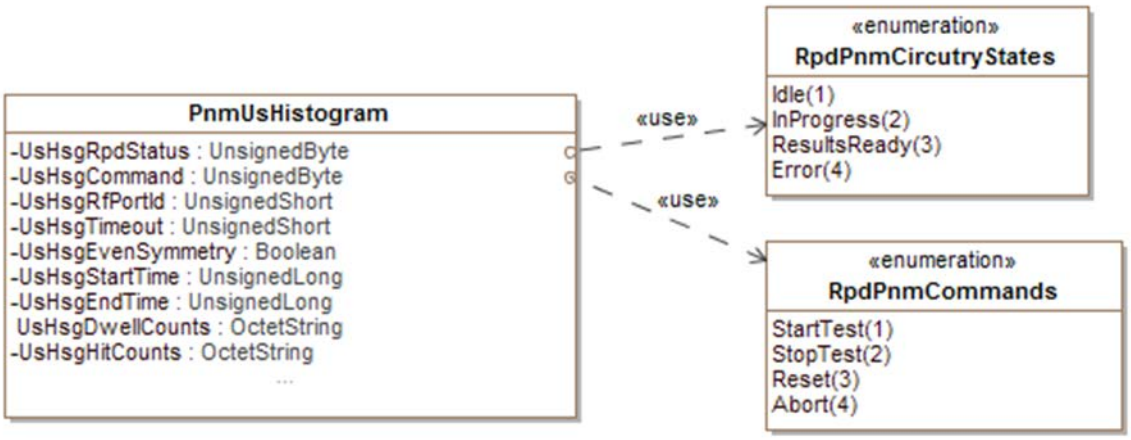

Figure 24 - GCP Objects used in US Histogram

Figure 24 shows the GCP objects used to facilitate Upstream Histogram control protocol. The object UsRfPort uniquely identifies the RPD's upstream RF port on which the histogram is collected. The UsHsgRpdStatus is a read-only object through which the CCAP Core can determine the current state of the RPD test histogram collection circuitry. The protocol defines four states of the RPD's upstream histogram collection circuity: "Idle", "InProgress", "ResultsReady" and "Error". The RPD reports the value "InProgress" while the test is running and a value "ResultsReady" when the test is complete. The object UsHsgCommand permits the CCAP Core to instruct the RPD to start, stop, reset and abort the histogram collection. The RPD reports "True" in the object UsHsgEvenSymmetry if it supports even symmetry of the histogram. The object UsHsgTimeout allows the CCAP Core to set a time limit (in seconds) for capturing histogram data. The RPD reports the start and end of collection through objects UsHsgStartTime and UsHsgEndTime. The object UsHsgEndTime reports a value of zero while the test is running. The CCAP may read these objects at any time when the test is running.

The RPD reports the histogram results through objects UsHsgDwellCounts and UsHsgHitCounts. The results are in the form of an array of 256 values, each value representing a single histogram bin in the form of an unsigned integer. If the RPD supports odd symmetry, the first bin (bin number zero) is not used and the RPD must report zero value for the number of dwells and hits.

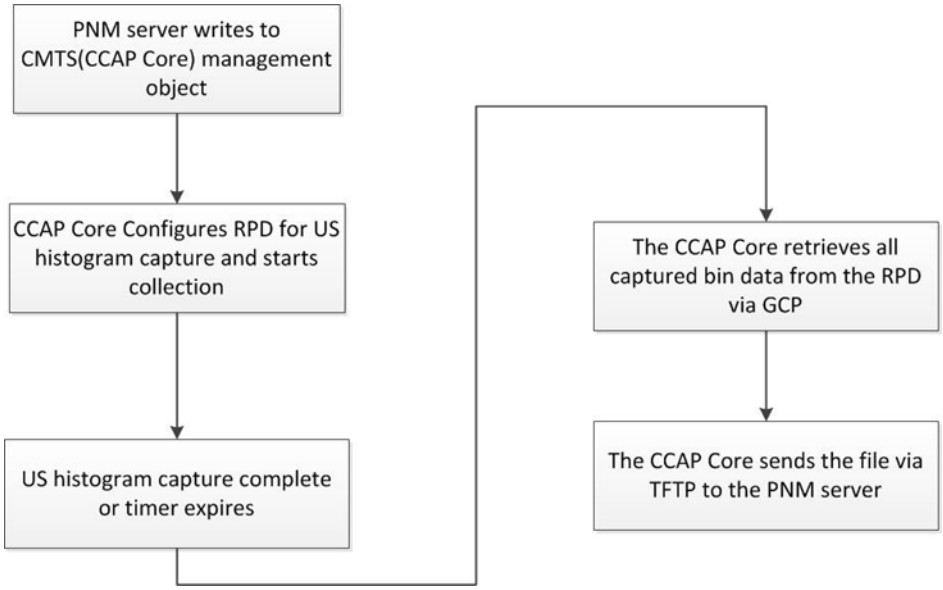

Figure 25 - US Histogram Capture Flow in R-PHY Architecture

Figure 25 shows the flow of US Histogram capture in an R-PHY architecture. The steps for US Histogram capture in the R-PHY environment are:

1. The PNM server writes to the CCAP Core objects, instructing it to start US Histogram capture.

2. The CCAP Core writes to the RPD to configure it for US Histogram capture through GCP. The CCAP Core selects the RPD's RF port by writing to UsHstRfPort object, an optional test timeout via UsHstTimeout object and starts the test by writing "StartTest" value to the UsHstCommand object. Next, the RPD starts collecting samples and generating the histogram. During the test run the CCAP Core can retrieve partial results from the RPD by reading UsHsgDwellCounts and UsHsgHitCounts objects.

3. The Upstream Histogram capture in the RPD is complete or a timer expires. If US histogram runs on a timer, the CCAP Core proceeds to Step 4 after the timer expires. In all other situations, the CCAP Core needs to check the status of the US histogram before proceeds to Step 4. The CCAP Core checks the RPD US histogram status by reading the "UsHsgRpdStatus" object via GCP. When the Status is "ResultsReady", the CCAP Core can proceeds to Step 4.

4. The CCAP Core retrieves the histogram from the RPD through GCP by reading UsHsgDwellCounts and UsHsgHitCounts objects.

5. The CCAP Core sends the PNM server the US Histogram data through TFTP.

Steps 1 and 5 are the same in both the I-CCAP and R-PHY architectures, permitting the same implementation of US Histogram capture function on the PNM server side in either architecture. The CCAP Core can abort the histogram collection at any time by writing "Abort" value to UsHstCommand object.

This specification assumes that the US Histogram captures are conducted sequentially. The CCAP Core MUST not initiate another US Histogram capture for the RPD before it completes or aborts the previous one. The CCAP Core MUST be capable of capturing one concurrent US Histogram per system. The RPD MUST be capable of capturing one concurrent US Histogram per device.

Annex A DEPI MTU (Normative)

A.1 L2TPv3 Lower Layer Payload Size

Typically, an interface calculates its default maximum payload size by asking the interface below it in the interface channel what is its maximum payload size and considering its own encapsulation. For example, by default, Ethernet has a frame size of 1518 (without VLANs). The Ethernet encapsulation is 18 bytes, leaving 1500 bytes of payload (MTU) for its upper layer. IP then subtracts the IP header size (typically 20 bytes) to arrive at 1480 bytes available to its upper layer. For D-MPT the remainder becomes 1472 bytes, because the Session Field and the L2TPv3 Data Session Header comprise 8 bytes. For PSP, the PSP header including the maximum PSP segment table size needs to be taken into account.

The CCAP Core and the RPD MUST support expanded Ethernet Frame sizes, up to 2000 bytes long, in compliance with [MULPIv3.1].

A.2 Maximum Frame Size for DEPI

This section documents the maximum frame size of the DEPI when a PSP pseudowire is used without fragmenting or concatenation.

Table 3 - MTU of DEPI (for PSP)

Field				Size
DEPI Frame	Ethernet Header			14 bytes
	802.1Q Header			4 bytes
	DEPI MTU	IPv4 Header		20 bytes
		IPv6 Header		40 bytes
		L2TPv3 Header		8 bytes
		DEPI-PSP Header*		6 bytes
		DOCSIS Frame	DOCSIS Header**	6-246 bytes
			Ethernet Header	14 bytes
			802.1Q Header	4 bytes
			Ethernet PDU	1500 or 2000 bytes
			Ethernet CRC	4 bytes
	Ethernet CRC			4 bytes
Total with PSP, no UDP, IPv4, no VLAN				**1570 to 1862 (or 2362)**

* A PSP header is 4 bytes plus 2 bytes for each segment. Only one segment is shown. (A D-MPT header is 4 bytes.)

** A typical DOCSIS header with BPI and no other extended headers is 11 bytes.

For simplicity, only one PSP segment is included in the above calculations. Additional segments are needed when PSP is concatenating or fragmenting. Note that a 2000 byte payload in a PSP frame could contain as many as 26 uncompressed TCP ACKs (64 byte Ethernet packets plus 6 to 11 bytes of DOCSIS overhead) which could create as many as 22 segments (first and last packets are fragmented) which would create a segment table size of 44 bytes, in addition to the standard 4 byte PSP header. For other payload types such as VoIP packets with high codec compression and with PHS disabled, or with larger MTUs, the number of segments could be even higher.

A.3 Path MTU Discovery

Path MTU Discovery relies on the fact that the network elements between the CCAP Core and the RPD all support this functionality [RFC 1191]. If these network elements do not support Path MTU Discovery then this mechanism cannot be used and the static configuration option should be used.

Path MTU Discovery (PMTUD) works when the IP path MTU between the CCAP Core and the RPD is less than the total IP datagram size generated when using the payload size negotiated during L2TPv3 session establishment, and the Don't Fragment (DF) bit is set in the IP header. If the CCAP Core sends packets larger than the network can support, then network elements between the CCAP Core and the RPD may generate an ICMP Destination Unreachable message with the code "Fragmentation needed and DF set" (ICMP Type 3 Code 4, also referred to as "Datagram Too Big" message), toward the source of the tunneled packet, if ICMP unreachables are allowed.

This ICMP error message includes at least the IP header and the next 8 bytes of the IP data (corresponding to the UDP header when using L2TPv3 over UDP, or to the Session ID and first 4 bytes of the L2SS when using L2TPv3 over IP) from the offending packet. The CCAP Core and the RPD should have a way to map the source and destination IP address contained in the IP header embedded in the ICMP data to an L2TP Control Connection. As defined in [RFC 1191], a "PMTU is associated with a path, which is a particular combination of IP source and destination address and perhaps a Type-of-Service (TOS)".

Upon successfully processing the ICMP Destination Unreachable message, the CCAP Core and RPD should reduce the Max Payload of all the sessions associated with the control connection mapped from the ICMP Destination Unreachable message to the size requested in the Next-Hop MTU field of the message. Both the Max Payload and the size contained in the Next-Hop MTU field express a Layer 3 payload of a Layer 2 frame, including the IP header and IP data.

The Max Payload MUST NOT be increased by receiving an ICMP Destination Unreachable message. The CCAP Core and RPD may periodically attempt to increase the Max Payload of the session to its negotiated maximum and restart this process in case the path through the network has changed and larger MTUs are allowed. This technique is described in [RFC 1191]. The Max Payload size learned through this process will never be greater than the negotiated maximum learned during session establishment. The Path MTU Discovery procedures for IPv6 are described in [RFC 1981].

Annex B GCP Usage (Normative)[18]

GCP (Generic Control Plane) is described in [GCP]. GCP is fundamentally a control plane tunnel that allows data structures from other protocols to be reused in a new context. This is useful if there is configuration information that is well defined in an external specification. GCP can repurpose the information from other specifications rather than redefining it. For example, MHAv2 uses GCP to reuse predefined DOCSIS TLVs for configuration and operation of the RPD. GCP has three basic features:

- Device management, such as power management;
- Structured access, such as TLV tunneling;
- Diagnostic access.

GCP defines the structured access using a combination of:

- 32 bit Vendor ID as defined in [Vendor ID];
- 16 bit Structure ID as uniquely defined by the vendor. For MHAv2, the default vendor ID is the CableLabs vendor ID of 4491 (decimal).

When GCP tunnels the data structures of another protocol, the syntax GCP(protocol name) can be used.

B.1 RPD Upstream Scheduler with GCP(DSx)

MHAv2 permits the upstream scheduler to be located either centrally in the CMTS Core or in the RPD. When the scheduler is located in the RPD, the CMTS Core needs to be able to add, change, and delete service flows in the remote upstream scheduler. The semantics for doing this are fully described in the DOCSIS DSA (Dynamic Service Flow Add), DSC (Dynamic Service Flow Change), and DSD (Dynamic Service Flow Delete) commands.

These commands are tunneled through GCP with the following parameters:

- Vendor ID = 4491 (CableLabs)
- Structure ID as defined in Table 4.

The DSx command headers are not needed, because GCP contains all header information and a transaction ID. The specific payload of the DSx commands that are used in the corresponding GCP commands are shown in Table 4. GCP does not have a separate ACK command since GCP is transported over a reliable transport protocol such as TCP. If the DSx-ACK TLVs are needed, they are carried over a second GCP Request Response pair.

Table 4 - GCP Encoding for the Upstream Scheduler

Structure ID	Function	GCP Message	GCP Payload
15	DSA-REQ	EDS-REQ	TLVs
16	DSA-RSP	EDS-RSP	Confirmation Code, TLVs
17	DSA-ACK	EDS-REQ	Confirmation Code, TLVs
17	n/a	EDS-RSP	No content
18	DSC-REQ	EDS-REQ	TLVs
19	DSC-RSP	EDS-RSP	Confirmation Code, TLVs
20	DSC-ACK	EDS-REQ	Confirmation Code, TLVs
20	n/a	EDS-RSP	No content
21	DSD-REQ	EDS-REQ	SFID, TLVs
22	DSD-RSP	EDS-RSP	Confirmation Code

[18] Revised per R-PHY-N-16.1573-3 on 9/7/16 by JB.

B.2 R-PHY Control Protocol

The following section defines the rules for the application of GCP as a Remote PHY control plane protocol. This set of rules is referred to as R-PHY Control Protocol or RCP.

RCP operates as an abstraction layer over the foundation of GCP protocol as defined in [GCP]. RCP provides the set of CCAP Core with to ability to remotely manage a set of objects, such as channels, ports, performance variables, etc.

RCP relies on the following GCP messages: Notify, Device Management and Exchange Data Structures. The encodings of the GCP messages are provided in tables below.

B.2.1 RCP over GCP EDS Message

Table 5 shows the encodings of the RCP over GCP EDS message.

Table 5 - RCP Encodings for GCP EDS Messages

Description	Length	Contents
Message ID	1 byte	6 (Exchange Data Structures Request)
Message Length	2 bytes	12 + N (length excludes three first bytes (Id +Len) of the header)
Transaction ID	2 bytes	Unique value
Mode	1 byte	0
Port	2 bytes	N/A
Channel	2 bytes	N/A
Vendor ID	4 bytes	4491 (IANA Enterprise Number assigned to CableLabs)
Vendor Index	1 byte	1
Message Body	N bytes	TLV encoded RCP Message

B.2.2 RCP over GCP EDS Response Messages[19]

The EDS Normal Response message shown in Table 6 has a format identical to the Request message (except Message ID == 7) and permits the inclusion of the TLV-encoded information.

Table 6 - RCP Encodings for GCP EDS Normal Response Messages

Description	Length	Contents
Message ID	1 byte	7 (Exchange Data Structures Request Normal Response)
Message Length	2 bytes	12 + N (length excludes three first bytes (Id +Len) of the header)
Transaction ID	2 bytes	Unique value, same as request
Mode	1 byte	0
Port	2 bytes	N/A
Channel	2 bytes	N/A
Vendor ID	4 bytes	4491 (IANA Enterprise Number assigned to CableLabs)
Vendor Index	1 byte	1
Message Body	N bytes	TLV encoded RCP Message

For each GCP request message the RPD MUST provide exactly one response message.

The EDS Error Response (Message Id == 8) format shown in Table 7 does not include TLV encoding information. This message can be used to communicate errors in those cases which are defined by the GCP specification [GCP].

[19] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB.

The types of errors which are not covered by GCP Error Response Message are conveyed in EDS Normal Response Message in TLV-encoded format.

Table 7 - RCP Encodings for GCP EDS Error Response Messages

Description	Length	Contents
Message ID	1 byte	135 (Exchange Data Structures Error Response)
Message Length	2 bytes	3
Transaction ID	2 bytes	Same as request
Exception code	1 byte	See section 6.4 of [GCP]

B.2.3 RCP over GCP Device Management Message

The RCP encodings of GCP Device Management messages are shown in Table 8.

Table 8 - RCP Encodings for GCP Device Management Messages

Description	Length	Contents
Message ID	1 byte	4 (Device Management)
Message Length	2 bytes	8
Transaction ID	2 bytes	Unique value
Mode	1 byte	Bit 7: 0 = Send normal response 1 = Suppress normal response Bit 6-0: Reserved. Set to 0.
Port	2 bytes	N/A
Channel	2 bytes	N/A
Command	N bytes	0 - Null 1 - Cold Reset 2 - Warm Reset 3 - Standby 4 - Wakeup 5 - Power-Down 6 - Power-Up 7 to 255 - Reserved

The RPD MUST set bit 7 of the Mode field to '1'.

B.2.4 RCP over GCP Notify Message

GCP Notify messages are sent from the RPD to the CCAP Core. RCP utilizes Event Code 1 and the TLV-encoded portion of the GCP Notify message. CCAP Core does not respond to Notify messages.

The RPD MUST set bit 7 to '1' and bit 6 to '1' in the Mode field. The RPD MUST set the value of the Event Code field to "1".

The RCP encodings of GCP Notify messages are shown in Table 9.

Table 9 - RCP Encodings for GCP Notify Messages

Description	Length	Contents
Message ID	1 byte	2 (Notify)
Message Length	2 bytes	8 + N (length does not include first 3 bytes of the message)
Transaction ID	2 bytes	Unique value, selected by the RPD.

Mode	1 byte	Bit 7: 0 = Send normal response 1 = Suppress normal response Bit 6: 0 = Event data is text 1 = Event data is raw Bit 5-0: Reserved. Set to 0.
Status	1 byte	0 - Null (default) 1 - Cold Reset 2 - Warm Reset 3 - Standby 4 - Wakeup 5 - Power-Down 6 - Power-Up 7 to 255 - Reserved
Event Code	4 bytes	1
Event Data	N bytes	TLV-encoded RCP message

B.2.5 RCP TLV Format, TLV Types and Nesting Rules[20]

The information carried in RCP protocol is formatted into TLV tuples. RCP operates with TLV format and usage rules which are similar to those defined in DOCSIS protocol. Each RPC TLV consists of a one byte long Type field, two byte long Length field and an optional, variable length Value field. The RPC TLV Type field can have the value of 1-255. The use of the value of "0" is reserved. The RPC TLV Length field denotes the total length of the Value field. The valid range for the Length field is 0-65535. When a TLV does not include the Value field, the Length field is set to zero. The RCP TLV format is presented in Figure 26.

Figure 26 - RCP TLV Format

As far as TLV Type is concerned, this specification defines two categories of TLVs: top level TLVs and sub-TLVs. The numbers representing TLV Types are assigned by method depending on the category of the TLV. A top level TLV is assigned a unique number from range 1-255. This specification refers to the top level TLV Types with a single number. Sub-TLVs are assigned Type numbers which are unique within the scope of their "parent" TLVs. Parent TLVs are those TLVs in which sub-TLVs are nested. Sub-TLV Types are represented in this specification as tuples, where the first number represents a top level TLV and consecutive numbers represent hierarchically nested sub-TLVs.

For example, the notation "50.19.9" refers to TLV SerialNumber, a sub-TLV with Type of 9, which is used to carry serial number of the RPD. The Serial Number is a sub-TLV of TLV RpdIdentification with type 19, which is used to convey information identifying RPD and is itself a sub-TLV of a top level TLV 50, RpdCapabilities.

As far as TLV Value field is concerned, there are two types of TLVs. Leaf TLVs and Complex TLVs. The Value field of a Leaf TLV contains a single data element. The encoding of the Value field of the leaf TLV varies; it depends on the TLV Type. Complex TLV are defined to have their Value field carry other, nested TLVs. A Complex TLV can carry a number of top level TLV or a number of sub-TLVs but never a mix of both categories.

Using these encodings, new parameters may be added which some devices cannot interpret. A CCAP Core or RPD which does not recognize a parameter type MUST skip over this parameter and not treat the event as an error condition.

[20] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB

B.2.6 RCP Message Structure[21]

The RCP Messages are embedded in a single TLV tuple. The value field of these TLV consists of multiple Sequence TLVs in the form {operation-TLV, Object Set-TLV}. The RCP protocol defines three operation types: "Read", "Write", and "Delete" and corresponding types for response messages. The definition of the managed objects, also referred to as information model or RCP schema is provided further in this specification.

The RCP TLV format imposes a size limit on RCP messages of 64 kB. RCP messages are never fragmented. When necessary, for example, if the volume of information exceeds RCP message limit (64 kB), the CCAP-Core can issue multiple messages. The sender of the RCP request message needs to anticipate that the response can be many times longer than the request. The GCP protocol does not allow for transmission of response message in multiple fragments. For this reason it is recommended to keep the size of request messages low.

B.2.7 RCP Messages Types[22]

The RCP protocol defines three message types. These messages, their TLV encoding, description and GCP usage are summarized in Table 10.

Table 10 - Summary of RCP Messages

Message Name	Message TLV Type	Description	GCP Mapping
IRA, Identification and Resource Advertising	01	An initial message exchanged after authentication in which the CCAP Core obtains all parameters identifying the RPD and its available resources.	Sent by CCAP Core in GCP EDS message.
REX, RCP Object Exchange	02	A message in which CCAP Core allocates or de-allocates resources and configures the resources in the RPD or requests information from the RPD, i.e., statistics or other status data.	Sent by the CCAP Core in GCP EDS message. Responded to by the RPD when operation is complete.
NTF, Notification	03	A message sent by the RPD to inform the CC about a specific event or a set of events.	Sent by the RPD in GCP Notify Message. CC does not respond to NTF messages.

B.2.8 RCP Protocol Rules

The CCAP Core can issue multiple RCP messages before it receives acknowledgement from the RPD. Each RPD MUST support a minimum of 16 outstanding messages per CCAP Core. A CCAP Core MAY issue a single IRA or REX message with a combination of read, write and delete tuples. The NTF issued by the RPD may only contain write tuples. A CCAP Core may issue a "Read" operation for a set individual objects (leaves) or object trees.

Responses to IRA and REX messages indicate the result of request processing with granularity of each {operation-TLV, Object Set-TLV} tuple. When the RPD response indicates a failure for a particular tuple, the RPD MUST make no change to the objects indicated in the tuple.

The RPD MUST respond to RCP request messages with one second of receiving the request message. The response messages sent by the RPD may be issued in a different order from the order of reception of request messages.

Since GCP operates over a reliable TCP connection) the protocol does not define explicit "acknowledgement" messages or other mechanism to deal with loss of individual messages.

B.2.8.1 *RCP Objects and TLVs23*

The RCP protocol operates on set of managed objects/TLVs sometimes referred to as ROTs (RCP Objects/TLVs). The ROTs are organized in a hierarchical tree. The top hierarchy consists of top level TLVs, which typically have a

[21] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB
[22] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB
[23] Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

complex structure and are referred as Container ROTs. Container ROTs typically represent a set of managed attributes. The bottom of the hierarchy is formed from Leaf ROTs, which are scalars or strings that represent a single managed attribute.

From a multiplicity perspective the RCP operates with two types of ROTs.

- Singleton ROTs, which have a single instance defined in the hierarchy. For example, "Capabilities" TLV is a Singleton ROT. Each RF channel and each RF Port in the RPD is represented by a separate Singleton ROT.
- Array ROTs can have multiple instances in and their definition includes one or more internal indexes. In Array ROTs, indexes are in the form of a small number sub-TLVs with defined ranges. The range of an index for each Array ROT is defined by this specification or by RPD's capabilities. For example, DedicatedToneConfig (TLV 61.7) is an array ROT. The range of indexes used with DedicatedToneConfig is defined by RPD's capability NumCwToneGens (TLV 50.21.1).

Both Singleton and Array ROTs can be defined as Leaf or Container ROTs.

B.2.8.1.1 Reading of Singleton ROTs

When CCAP Core issues a read request for a Singleton ROT, the RPD returns the entire content of the TLV sub-tree represented by the ROT. For example, when the CCAP Core issues a read request for the "Capabilities" TLV, the RPD returns, in response, all sub-TLVs of the "Capabilities" TLV including multiple instances of Array ROTs within the hierarchy of "Capabilities". Since the response includes the entire sub-tree, the size of response can be very large. The RCP protocol does not specify a method to limit the maximum size of the response. For this reason, the read requests need to be limited in order to not exceed the protocol limits (64 KB per TLV).

The CCAP Core can also select, for a read-request, a Singleton TLV representing a portion of the tree in the hierarchy, down to a leaf.

For example, the CCAP Core can issue a read request for Capabilities.RpdIdentification (Container ROT, TLV 50.19) and in response, the RPD needs to return the entire content of that sub-TLV. The response will contain 16 sub-TLVs.

In another example, the CCAP Core can issue a read request for Capabilities.RpdIdentification.BootRomVersion (Leaf ROT, TLV 50.19.6), and, as the result, the RPD needs to return just this one leaf sub-TLV value.

B.2.8.1.2 Reading of Array ROTs

The RCP protocol specifies a method by which read requests to Array ROTs can be issued with selection of a number and a range of instances of ROTs.

When a read request is issued for Array ROT, the index (or a set of indexes) within the ROT defines the starting index within the Array ROT. When a read request is issued for an Array ROT and indexes sub-TLVs are not included in the request, then the RPD assumes the lowest (or starting) indexes values.

A top-level "ReadCount" (TLV 26) is defined to specify how many instances of the ROT are to be returned in read-response. ReadCount TLV has an unsigned short value permitting index ranges from 0-65535. ReadCount TLV is placed immediately before the TLV specifying the starting Array ROT instance. When ReadCount TLV is omitted, the RPD returns just one ROT instance. For multidimensional arrays, the array entries are returned in the order of consecutive indexing.

For example, if an array ROT "X" has two indexes A(0-5) and B(0-6), a Read request for 5 instances starting from indexes A=2, B=3, then the RPD returns instances of "X" in the following order: X [A=2,B=3], X [A=2.B=4], X [A=2.B=5], X [A=2.B=6], X [A=3.B=0].

The CCAP Core can issue a Read request with ReadCount TLV value which is larger than the number of instances actually supported by the RPD. In such case the RPD returns all supported instances of the requested Array ROT.

B.2.9 Protocol Extensibility[24]

This section will be written for a future version of this specification.

B.2.10 Protocol Versioning

The RCP protocol uses versioning as the primary means for future extensibility. The initial RCP protocol version defined by this specification is "1.0". Future versions of this specification may define new RCP protocol versions with additional capabilities or protocol options. During the initialization the CCAP Core will read the RPD's capabilities, including the set of RCP protocol versions supported by the RCP via the IRA message. The CCAP will then select the highest RCP protocol version that both the CCAP Core and the RPD can support and instruct the RPD to use the selected version.

B.2.11 Information Model Extensibility

The RPHY information model/schema is versioned separately from the protocol. The method for schema version selection is similar to the protocol version selection. The initial RCP information schema version defined by this specification is "1.0" Future versions of this specification may define new RCP information schema versions. For each version of the schema this specification will define a set of mandatory objects and a set of optional objects organized in sets, referred to as features. During initialization, the CCAP Core will read which schema features the RPD supports in the IRA message. The CCAP Core will also let the RPD know (write) which versions of the schema and which features it supports to control objects sent in Notify messages.

The CCAP Core MUST convey in RCP protocol only those objects that the RPD supports. RPD MUST convey in RCP protocol only those objects that the CCAP Core supports. These requirements are not applicable to vendor specific extensions.

B.2.12 Vendor Specific Extensions[25]

The RCP protocol permits for exchange of vendor specific information by defining a method for inclusion of vendor specific TLVs. Vendor specific TLVs are complex TLVs with a Type of "Vendor-Specific". The first sub-TLV of a vendor specific TLV is the TLV identifying the vendor with length of 2 and the value field containing the vendor's Private Enterprise Number (http://www.iana.org/assignments/enterprise-numbers/enterprise-numbers). A vendor specific TLV includes one or more vendor defined sub-TLVs. The definition of the formats and the usage of these sub-TLVs are outside of the scope of this specification.

An example of vendor specific TLV is provided below.

```
{T= Vendor-SpecificExtension, Length: variable (minimum)
    {T = Vendor Id, L = 2, V = Vendor ID: Enterprise number identifying vendor}
    {
        A sequence consisting of one or more vendor specific TLVs.
    }
}
```

Vendor-specific TLVs are ignored by RPDs and CCAP Cores which do not recognize vendor id.

B.2.12.1 *Vendor-specific Pre-configuration (VSP)*[26]

Vendor-specific Pre-configuration (VSP, Length: 0.. 1024 bytes) is an arbitrary set of TLVs written by the CCAP Core to the RPD before any other RPD configuration. One possible use is a set of vendor-specific extension TLVs to configure "QAM blocks" of downstream SC-QAM channels that the vendor requires to have the same base power, modulation, and/or interleave. Each RPD vendor determines its own VSP TLV setting appropriate for an MSO's intended requirements and communicates to that MSO:

The 2-byte RPD Vendor ID;

A "VsoSelector" string reported by the RPD as a capability; and

[24] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB
[25] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB
[26] Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

The hexadecimal representation of a VSP Setting corresponding to that VSP Selector.

The VsoSelector is a DisplayString 0.. 16 byte long chosen by the RPD vendor to select among multiple possible VSP Settings for that vendor configured at the CCAP Core.

The VSP Setting consists of up to 1024 bytes of TLVs for a GCP REX "Write" message that follows the "Write" operation TLV. The VSP Setting contents are opaque to the CCAP Core.

In the Identification and Resource Advertising (IRA) phase of GCP establishment, a CCAP Core reads from an initializing RPD its “Vendor ID” and "VSP Selector" capability objects. When the CCAP Core initializes an RPD from cold-start and contains a VSP Mapping of that RPDs “Vendor ID” and “VSP Selector”, the CCAP Core MUST write the mapped VSP Setting to the RPD via a single REX WRITE message before any other configuration objects are written to the RPD.

The CCAP Core MUST reject the initialization of an RPD that fails to acknowledge the write of its mapped VSP Setting. For testing of this requirement, the RPD MUST reject a REX write to a Vendor Specific Extension TLV with the Enterprise ID of 0x0000, which is reserved by IANA.

The RPD MUST store its VSP setting in a volatile manner, resetting any vendor-proprietary state configured with VSP to a factory default value on each GCP cold-start. An RPD MUST maintain its initial VSP Setting on a GCP warm-start.

An RPD may not implement VSP, in which case it MUST report its "VSP Selector" capability as an empty (zero-length) string.

Only the Principal CCAP Core writes a VSP to an RPD. An Auxiliary CCAP Core MUST NOT write a VSP, even if it matches a Vendor ID and VSP Selector mapped on the Auxiliary CCAP Core.

B.2.13 Inclusion of DOCSIS Messages[27]

The CCAP Core can include in RCP certain messages describing the majority of the parameters of US TDMA and OFDMA channels and DS OFDM channels. These messages are transmitted in the form of a TLVs in REX messages.

The RPD MUST support the reception of three types of DOCSIS messages, including UCD, OCD and DPD Messages as the means for configuration of selected DOCSIS channels for which these messages provide description. The RPD MUST decode these messages using rules defined in DOCSIS MULPI specifications in order to configure selected channel resources.

The CCAP Core MUST support configuration of a downstream OFDM channel by sending an OCD message to the RPD via GCP.

The CCAP Core MUST support configuration of a downstream OFDM profile by sending a DPD message to the RPD via GCP.

The CCAP Core MUST support configuration of an upstream channel by sending a UCD message to the RPD via GCP.

When sending multipart DOCSIS messages, the CCAP Core MUST include all DOCSIS parts message in a single GCP/RCP message.

Two examples of RCP messages containing an embedded DOCSIS message are provided in Section B.2.15.4.

B.2.13.1 *Dynamic Change Procedures*

[MULPIv3.1] defines dynamic change procedures for upstream channel parameters and downstream OFDM channel profiles. In an integrated CMTS these procedures involve precise coordination of timing of operations between the CMTS and CMs. An R-PHY System complies with relevant requirements of [MULPIv3.1] as well as with additional protocol rules defined to permit an orderly transition from the old parameter values to the new values

[27] Revised per R-PHY-N-15.1403-3 on 1/11/16 and per R-PHY-N-16.1573-3 on 9/7/16 by JB. Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

between the CCAP Core and RPDs. The following requirements have been established to allow seamless implementation of upstream channel and downstream OFDM profile change procedures in the Remote PHY system.

B.2.13.1.1 UCD Change Procedure

When requesting configuration changes to an upstream channel, the CCAP Core needs to ensure that the RPD receives all necessary configuration information including the time when the change to the respective parameters is to be applied in RPD's upstream burst receiver. The CCAP Core also needs to ensure that the RPD has sufficient time to process the new configuration information before it is applied in processing upstream bursts.

There are two attributes the CCAP Core sends to the RPD to initiate the UCD change procedure:

- A UCD message with incremented UCD change count.
- A 32-bit DOCSIS timestamp indicating the UCD configuration change time. The 32-bit DOCSIS timestamp points to Alloc Start Time in first MAP message with changed UCD count.

The CCAP Core MUST ensure that the 32-bit DOCSIS timestamp points to the interval corresponding to the start of the first grant in the MAP with incremented UCD Change Count. As required by [MULPIv3.1] the first grant in MAP with incremented UCD Change Count is a data grant to the Null SID.

The RPD UCD Advance Time is defined as a difference between the time of completion of transmission of the GCP message with UCD message and the time of transmission of the first bit of the first MAP using the new UCD. The CCAP Core calculates the RPD UCD Advance Time as a sum of two intervals:

RPD UCD Processing Time. This interval is equivalent to CM UCD processing time defined in [MULPIv3.1] however its duration can be longer. The RPD advertises its required RPD UCD Processing time via Capabilities. The maximum value of the RPD UCD Processing time is 50 msec. The minimum RPD UCD Processing time is equal to CM UCD processing time (1.5 msec for each changed SC-QAM channel or 2.0 msec for each changed upstream OFDMA channel) defined in [MULPIv3.1].

Estimated transmission propagation delay from the CCAP Core to the RPD. CCAP Core estimates the transmission propagation delay based on DLM measurements and other methods which are outside of the scope of this specification.

The CCAP Core MUST complete transmission of the UCD message and the 32-bit timestamp to the RPD via GCP at minimum RPD UCD Advance Time ahead of the scheduled UCD configuration change time.

The RPD capabilities also advertise the RPD UCD Change Null Grant Time, which specifies the minimum amount of time the RPD needs to program its burst receiver registers during the first MAP with incremented UCD change time. The maximum value of the RPD UCD Change Null Grant Time is 4 msec for each changed channel. The minimum value of the RPD UCD Change Null Grant Time is defined in [MULPIv3.1].

When performing UCD change procedure, the CCAP Core MUST transmit the first MAP message with incremented UCD change count in which the first interval is a data grant to the Null SID that has a minimum length of the RPD's advertised RPD UCD Change Null Grant Time for each simultaneously changed channel.

When performing UCD change procedure, the CCAP Core MAY transmit the first MAP message with incremented UCD change count in which the first interval is a data grant to the Null SID that is longer than the RPD's advertised RPD UCD Change Null Grant Time for each simultaneously changed channel.

The RPD determines the UCD change time through one of the two methods outlined below:

- The RPD can examine the MAP stream sent on the MAP pseudowire and apply the changes to the channel's parameters when the RPD detects the Configuration Change Count incremented in the processed MAP stream for the channel. This method is similar to the UCD change procedure supported by DOCSIS CMs. When RPD supports this method, the RPD does not have to take advantage of the 32-bit DOCSIS timestamp supplied by the CCAP Core via GCP.
- The RPD determines the upstream channel change time from the 32-bit DOCSIS timestamp explicitly signaled by the CCAP Core via GCP.

The selection between these methods is left to RPD's implementation choice.

B.2.13.1.2 OFDM Profile Change Procedure

[MULPIv3.1] defines downstream OFDM profile change procedure for I-CCAP. In R-PHY system, I-CCAP responsibilities are divided between the CCAP Core and the RPD. Figure 27 shows the comparison of the OFDM Profile change procedures when performed in an I-CCAP and in an R-PHY system.

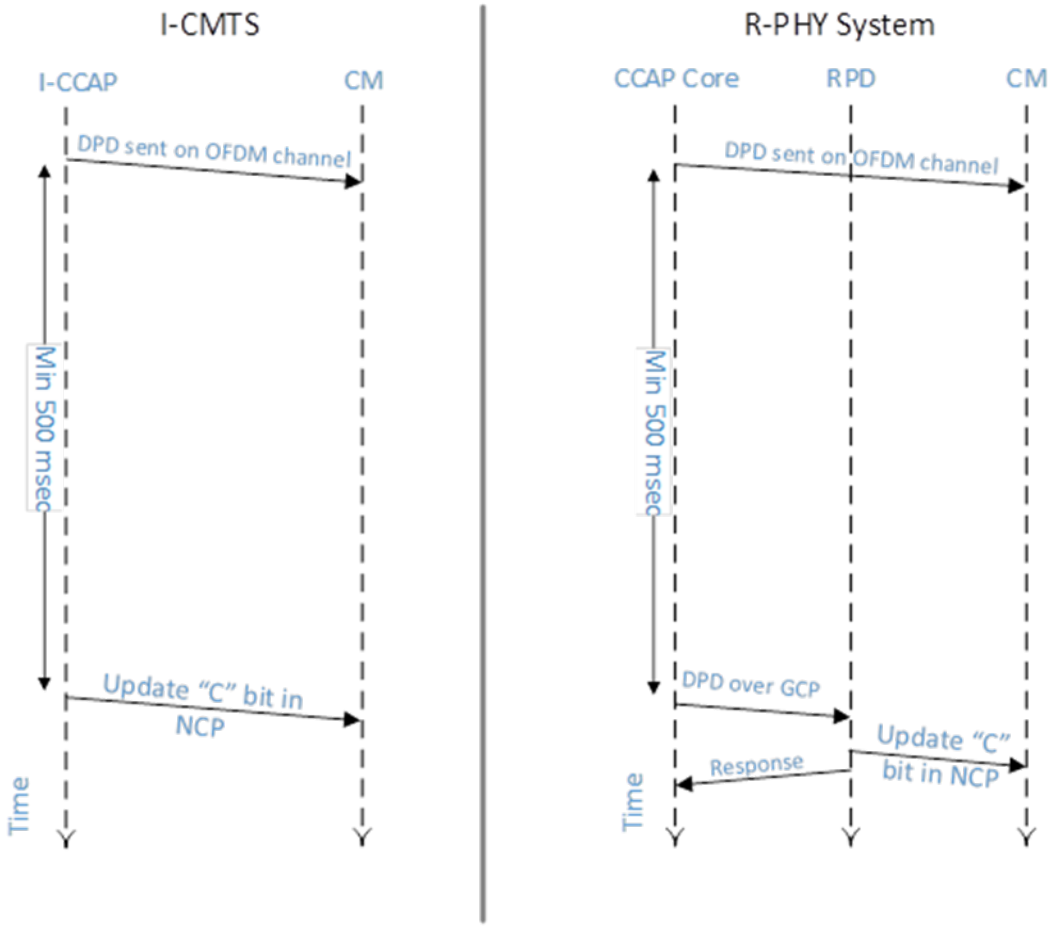

Figure 27 - Comparison of OFDM Profile Change Procedures between I-CCAP and R-PHY System

In an R-PHY system, the CCAP Core first sends the updated DPD with incremented Configuration Change Count to the CMs on the OFDM channel. After Profile Advance Time, the CCAP Core requests that the RPD performs the OFDM Profile change by sending the same DPD message to the RPD over GCP. After receiving a DPD message from the CCAP Core, the RPD updates its OFDM modulator with new profile parameters and subsequently updates the NCP "C" bit to match the LSB of the DPD change count. Once this operation is completed, the RPD responds to the GCP message.

The CCAP Core MUST send the updated DPD message to the RPD a minimum Profile Advance Time after transmitting the same DPD message on the OFDM channel. Profile Advance Time is defined in [MULPIv3.1] and has a value of 500 msec. This requirement is intended to ensure that the RPD does not perform the OFDM profile change before the affected CMs have sufficient time to act on the DPD message.

The RPD MUST complete downstream OFDM profile change within 100 msec of the reception of the request from the CCAP Core.

B.2.13.2 *OFDM Channel Configuration*

The CCAP Core configures the parameters of a downstream OFDM channel in the RPD by sending to it an OCD message via GCP. Unlike the parameters in the UCD or the DPD message which can be changed dynamically, the assignment of parameters communicated via OCD message is generally considered a static. Any changes to the parameters communicated by the OCD message can disrupt the traffic on the channel and cause the CMs to lose the ability to receive the data sent on the channel.

B.2.14 Event Reporting[28]

The GCP/RCP protocol facilitates reporting of events by the RPD to the CCAP Core and configuration of event reporting in RPD's Local Event Log. There are three methods by which event reports generated by the RPD can be received by the CCAP Core.

The primary method for delivery of event reports is by GCP/RCP Notify messages. The Principal CCAP Core can configure the RPD to send event reports to the Principal CCAP Core by enabling selected event priority levels in RpdGlobal.EvCfg.EvControl attributes and by enabling transport of event reports in Notify messages via RpdGlobal.EvCfg.EvControl.NotifyEnable attribute. Two examples of a Notify message encoding can be found in Section B.2.15.5.

The RPD only reports newly generated event counts to the CCAP Core. For example, if an event has occurred five times and the RPD has previously sent a report for this event, with EvCount attribute set to three, then the RPD sends an event report indicating only two new occurrences of the event (EvCount set to 2). In another example, if an event has occurred three times and the RPD has not yet reported this event, the RPD sends an event report indicating all three occurrences (EvCount is set to three).

The transmission of event reports to the CCAP Core via Notify message is subject to throttling. The RCP supports several attributes to control the throttling. These attributes are modeled after [RFC 4639].

When the RPD is configured to send event reports to the Principal CCAP Core but does not have connectivity to the Principal CCAP Core, or when the RPD is not enabled to send event reports via Notify messages, then the RPD stores new event reports in the Pending Event Report Queue. The Pending Event Report Queue is intended to operate as a temporary storage for event reports intended for the CCAP Core, when Notify message transport is not available or when it is disabled. The RPD MUST aggregate event reports in the Pending Event Report Queue. When the RPD generates two (or more) events reports for the same EvId, then the RPD combines them into a single report that contains the EvCount, which is the sum of individual event counts and a single set of EvFirstTime and EvLastTime timestamps.

The CCAP Core can read the Pending Event Report Queue via GCP/RCP. This method can be utilized, for example, during the connection initialization to prevent the RPD from uncontrolled flooding of the CCAP Core with event reports that may have been generated during or prior to RPD's initialization. When the CCAP Core reads event reports from the Pending Event Report Queue, the RPD delivers the reports in the order they have been stored in the queue. The oldest report is delivered first. Any report read by the Principal CCAP Core is removed from the Pending Event Report Queue by the RPD. The CCAP can clear the Pending Report Queue.

The RPD MUST preserve the content of the Pending Event Report Queue across reboots in its non-volatile memory. The RPD MUST support Pending Report Queue with a minimum of 20 entries. When the Pending Event Report Queue is full, and the RPD needs to report with a new event, the RPD SHOULD discard the oldest event report and insert the new event report.

The CCAP Core has the ability clear the Pending Event Report Queue instead or reading it.

The CCAP Core can also configure the RPD to store event reports in RPD's Local Event Log. The CCAP Core can directly read RPD's Local Event Log or clear it.

Additional information about R-PHY events, including the definition of standard events, the format of event reports generated by the RPD and their handling by the CCAP Core can be found in [R-OSSI].

[28] Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

B.2.15 RCP Message Examples[29]

B.2.15.1 *RCP Rex Message Request Example*

The following example presented below represents a message with a single "Read" operation for a set of statistical counters for two upstream channels. Curly braces "{" and "}" denote the boundaries of TLVs. Note, that the outer envelope (GCP EDC Request) is not shown.

```
{ T = REX, L= 66, V =                          ; top-level "container" type
   { T = Sequence, L = 63, V =                           ; a seq. of TLVs starting
with oper.
      { T = SequenceNumber, L = 2, V = 21 }
      { T = Operation, L = 1, V = Read }
      { T = RfChannel, L = 24, V =             ; L = 6 + 3*3 = 15
          { T = RfChannelSelector, L = 12, V =
             { T = RfPortIndex, L = 1, V = 6 }
             { T = RfChannelType, L = 1, V = 1 }
             { T = RfChannelIndex, L = 1, V = 7}
          }
          { T = TotalCW, L = 0  }
          { T = UncorrectedCW, L=0 }
          { T = CorrectableCW, L=0 }
      }
      { T = RfChannel, L =  24, V =
          { T = RfChannelSelector, L = 12, V =
             { T = RfPortIndex, L = 1, V = 2 }
             { T = RfChannelType, L = 1, V = 1 }
             { T = RfChannelIndex, L = 1, V = 7}
          }
          { T = TotalCW, L = 0 }
          { T = UncorrectedCW, L=0 }
          { T = CorrectableCW, L=0 }
      }
   }
}
```

B.2.15.2 *RCP REX Message Normal Response Example*

The message below represents a successful (no error) REX Response for stats counters for two upstream channels.

This is a response to the request outlined in Section B.2.15.1. As in previous examples, the outer envelope (GCP EDC Normal Response) is not shown.

```
{ T = REX, L = 90, V =                                                  ; top-level
"command" type
    { T = Sequence, L = 87, V =
      { T = SequenceNumber, L = 2, V = 21 }
      { T = Operation, L = 1, V = ReadResponse }
      { T = RfChannel, L =  36, V =
          { T = RfChannelSelector, L = 12, V =
             { T = RfPortIndex, L = 1, V = 6 }
             { T = RfChannelType, L = 1, V = 1 }
             { T = RfChannelIndex, L = 1, V = 7}
           }
          { T = TotalCW, L = 4, V = 32-bit counter}
          { T = UncorrectedCW, L = 4, V = 32-bit counter }
          { T = CorrectableCW, L = 4, V = 32-bit counter }
      }
      { T = RfChannel, L =  36, V =
          { T = RfChannelSelector, L = 12, V =
             { T = RfPortIndex, L = 1, V = 2 }
```

[29] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB

```
               { T = RfChannelType, L = 1, V = 1 }
               { T = RfChannelIndex, L = 1, V = 7}
             }
            { T = TotalCW, L = 4, V = 32-bit counter }
            { T = UncorrectedCW, L = 4, V = 32-bit counter }
            { T = CorrectableCW, L = 4, V = 32-bit counter }
      }
   }
}
       }
```

B.2.15.3 *RCP Rex Message Error Response Example*

The example shown below represents a REX Response message for stats counters for two upstream channels. This is a response to the request outlined in Section B.2.15.1. As in the previous examples, the outer envelope (GCP EDC Normal Response) is not shown. The values of the response code (rspCode) are to be determined.

```
{ T = REX, L= 87, V =
   { T = Sequence, L = 85, V =
   { T = SequenceNumber, L = 2, V = 21 }
      { T = Operation, L = 1, V = ReadResponse }
      { T = ResponseCode, L = 1, V = rspCode }  ;A Status TLV , only required on an
error
                                                ;if omitted, operation was
successful
      { T= ErrorMessage, L = 15, V = ”Unknown channel = 6.1.7” }
                                                ;Optional RPD vendor spec. msg for
the log
      { T = RfChannel, L = 24, V =
         { T = RfChannelSelector, L = 12, V =
            { T = RfPortIndex, L = 1, V = 6 }
            { T = RfChannelType, L = 1, V = 1 }
            { T = RfChannelIndex, L = 1, V = 7}
         }
         { T = TotalCW, L = 0 }
         { T = UncorrectedCW, L = 0}
         { T = CorrectableCW, L = 0}
      }
      { T = RfChannel, L = 24, V =
          { T = RfChannelSelector, L = 12, V =
             { T = RfPortIndex, L = 1, V = 2 }
             { T = RfChannelType, L = 1, V = 1 }
             { T = RfChannelIndex, L = 1, V = 7}
          }
          { T = TotalCW, L = 0}
          { T = UncorrectedCW, L = 0}
          { T = CorrectableCW, L = 0}
      }
   }
}
```

B.2.15.4 *Examples of an Embedded DOCSIS Message*[30]

The example shown below represents a REX Request Message, in which the CCAP Core communicates to the RPD the content of a DOCSIS message.

```
{ T = REX, L= nn + 40, V =                  ; top-level “container” type
   { T = Sequence, L = nn + 37, V =      ; nn is the length of the DOCSIS Message
      { T = SequenceNumber, L = 2, V = 0211 }
      { T = Operation, L = 1, V = Write }
```

[30] Revised per R-PHY-N-16.1573-3 on 9/7/16 by JB. Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

```
         { T = RfChannel, L =  nn + 25, V =
            { T = RfChannelSelector, L = 12, V =
               { T = RfPortIndex, L = 1, V = 2 }
               { T = RfChannelType, L = 1, V = 5 } ;ATDMA channel
               { T = RfChannelIndex, L = 1, V = 7}
            }
            { T = DocsisMsg, L = nn, V = "A Hex String with a complete DOCSIS message"
   }
            { T = StartingMinislot, L = 4, V = 0x11223344 } ; (66.11)
         }
      }
   }
```

The example shown below represents a REX Request Message, in which the CCAP Core writes a multipart UCD message to the RPD.

```
   { T = REX, L= variable, V =                  ; top-level "container" type
      { T = Sequence, L = nn1+nn2+nn3 +43, V =  ; a seq. of TLVs starting with oper.
         { T = SequenceNumber, L = 2, V = 21 }
         { T = Operation, L = 1, V = Write }
         { T = RfChannel, L = Variable, V =
            { T = RfChannelSelector, L = 12, V =
               { T = RfPortIndex, L = 1, V = 6 }
               { T = RfChannelType, L = 1, V = 6 } ; OFDMA channel
               { T = RfChannelIndex, L = 1, V = 7}
            }
            { T = DocsisMsg, L = nn1, V = UCD1-part1 }
            { T = DocsisMsg, L = nn2, V = UCD1-part2 }
            { T = DocsisMsg, L = nn3, V = UCD1-part3 }
            { T = StartingMinislot, L=4, V = Change Time } ; TLV (66.11)
         }
      }
   }
```

B.2.15.5 *Examples of a Notify Message*[31]

The first example shows a generic encoding of a Notify message.

```
{ T = NTF, L= N, V =                          ; top-level "container" type
   { T = Sequence, L = N, V =                 ; a seq. of TLVs starting with oper.
      { T = SequenceNumber, L = 2, V = 4567 }  ; RPD selects sequence number
      { T = Operation, L = 1, V = Write }
      {
        A Top Level TLV containing notification information.
      }
   }
}
```

The second example shown below represents a Notify Message in which the RPD sends an event report to the CCAP Core. The event report shown in the example describes a single occurrence of the event with id 66070415. For this reason the event report includes the EvFirstTime attribute and does not include EvLastTime attribute.

```
{ T = NTF, L= 208, V =                  ; top-level "container" type
```

[31] Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

```
  { T = Sequence, L = 205, V = 123         ; a seq. of TLVs starting with oper.
     { T = SequenceNumber, L = 2, V = 2507 } ; RPD assigns Sequence Number
     { T = Operation, L = 1, V = Write }
     { T = EventNotification, L = 193, V =
        { T = EvFirstTime, L = 11, V = '07DE0A060F0000002D0600' } ; 2014-10-6,
15:00:00.0, -6:00
        { T = EvCounts, L = 4, V = 1 }
        { T = EvLevel, L = 1, V = 4 }     ; Error Event
        { T = EvId, L = 4, V = 66070415 } ; Code File Co-Signer CVS Validation Failure
        { T = EvText, L = 158, V = "Code File Co-Signer CVS Validation Failure;SW
File:RPD-vendor-X-6789;Server:10.11.34.105;RPD-MAC=00:22:ce:03:f4:da;CCAP-
MAC=00:15:20:00:25:ab;RPD-MHA-VER=1.0;" }
      }
   }
}
```

The third example represents a Notify Message in which the RPD sends an event report indicating five occurrences of the event with id 66070415. For this reason the event report includes both the EvFirstTime and EvLastTime attributes.

```
{ T = NTF, L= 222, V =                   ; top-level "container" type
  { T = Sequence, L = 219, V = 123         ; a seq. of TLVs starting with oper.
     { T = SequenceNumber, L = 2, V = 2507 } ; RPD assigns Sequence Number
     { T = Operation, L = 1, V = Write }
     { T = EventNotification, L = 207, V =
        { T = EvFirstTime, L = 11, V = '07DE0A060F0000002D0600' } ; 2014-10-6,
15:00:00.0, -6:00
        { T = EvLastTime, L = 11, V = '07DE0A060F0816002D0600' } ; 2014-10-6,
15:08:22.0, -6:00
        { T = EvCounts, L = 4, V = 5 }
        { T = EvLevel, L = 1, V = 4 }     ; Error Event
        { T = EvId, L = 4, V = 66070415 } ; Code File Co-Signer CVS Validation Failure
        { T = EvText, L = 158, V = "Code File Co-Signer CVS Validation Failure;SW
File:RPD-vendor-X-6789;Server:10.11.34.105;RPD-MAC=00:22:ce:03:f4:da;CCAP-
MAC=00:15:20:00:25:ab;RPD-MHA-VER=1.0;" }
      }
   }
}
```

B.3 GCP Connection Initialization Sequence[32]

The RCP initialization sequence is shown in Figure 28.

[32] Revised per R-PHY-N-15.1403-3 on 1/11/16 by JB.

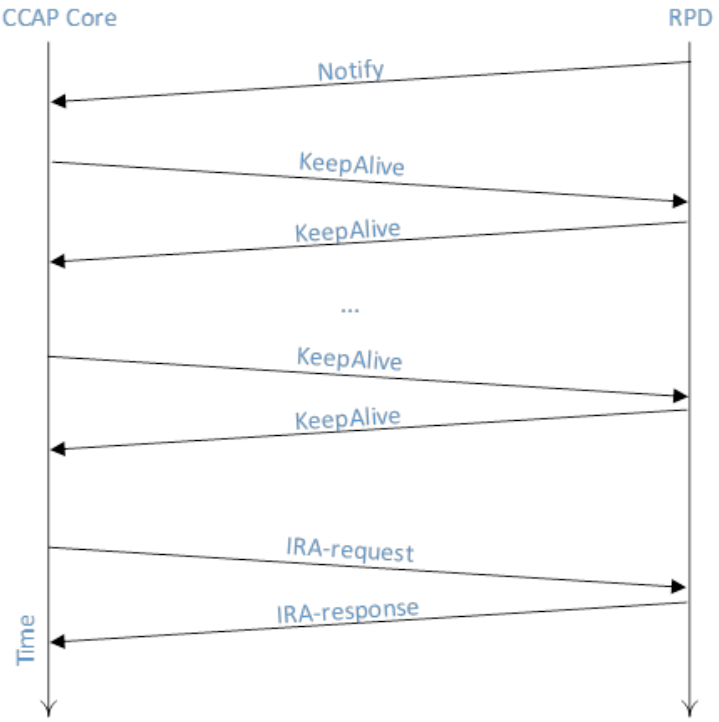

Figure 28 - RCP Initialization Sequence

After establishing the TCP connection, the RPD first sends a NTF message to the CCAP Core to allow the CCAP Core to identify the RPD. The RPD MUST include the following RPD Identification TLVs in the initial notification message:

VendorName, 50.19.1

VendorId, 50.19.2

ModelNumber, 50.19.3

DeviceMacAddress, 50.19.4

CurrentSwVersion, 50.19.5

BootRomVersion, 50.19.6

DeviceDescription, 50.19.7

DeviceAlias, 50.19.8

SerialNumber, 50.19.9

RpdRcpProtocolVersion, 50.19.14

RpdRcpSchemaVersion, 50.19.15

DeviceLocation, 50.24

Based on the received information in the initial NTF message, the CCAP Core can redirect the RPD to another CCAP Core (or a set of CCAP Cores) as described in Section 6.6.2.6. The CCAP Core redirects the RPD by sending to the RPD an IRA message with RpdRedirect TLV which includes an ordered list of IP addresses of CCAP Cores to contact next. The CCAP Core can delay the process of redirecting the RPD for up to 60 seconds to allow the CCAP Cores to prepare to service the RPD. When the RPD receives a redirect request, it MUST send a response to the IRA message and tear down the TCP connection to the redirecting CCAP Core and immediately attempt to connect to the CCAP Core it is redirected to.

Figure 28 also shows the KeepAlive messages exchanged between the RPD and CCAP Core. This specification recommends exchange of KA message every one second.

If the CCAP Core does not redirect the RPD, it may proceed further by reading other RPD's capabilities via the IRA message, and later configuring the RPD via REX messages.

NOTE: The initialization sequence does not include capability negotiation in this version of the specification as both the RPD and the CCAP-Cores are required to support version "1.0" for the RCP protocol and the version "1.0.x" for the RCP schema.

B.4 Summary GCP TLV Encodings[33]

B.4.1 RCP Top Level TLVs

Table 11 displays the summary of top level TLVs which provide the outer encapsulation to the TLV encoded data in the RCP messages.

Table 11 - RCP Commands

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
IRA	Complex TLV	1	variable		IRA Message
REX	Complex TLV	2	variable		REX Message
NTF	Complex TLV	3	variable		Notify Message

B.4.2 General purpose TLVs

Table 12 shows the list of general purpose TLVs used in RCP protocol.

Table 12 - RCP Top Level TLVs[34]

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
Sequence	Complex TLV	9	variable		
SequenceNumber	UnisignedShort	10	2		
Operation	UnsignedByte	11	1		Enumerated command and response codes
RfChannelSelector	Complex TLV	12	12		
RfPortIndex	UnsignedByte	12.1	1		
RfChannelType	UnsignedByte	12.2	1		
RfChannelIndex	UnsignedByte	12.3	1		
RfPortSelector	Complex TLV	13	8		
RfPortIndex2	UnsignedByte	13.1	1		
RfPortType	UnsignedByte	13.2	1		
EnetPortIndex	UnsignedByte	14	1		
RpdGlobal	Complex TLV	15	variable		
RfChannel	Complex TLV	16	variable		
RfPort	Complex TLV	17	variable		

[33] Added per R-PHY-N-15.1360-4 on 9/22/15 and per R-PHY-N-15.1403-3 on 1/11/16 by JB.
[34] Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
EnetPort	Complex TLV	18	variable		
ResponseCode	UnsignedByte	19	1		
ErrorMessage	String	20	variable	1-255	Error msg for the log.
VendorSpecifc Extension	ComplexTLV	21	variable		
VendorId	UnsignedShort	21.1	2		
DocsisMsg	OctetString	22	variable		
DocsisTimestamp32	UnsignedInt	23	4		
DocsisTimestamp64	UnsignedLong	24	8		
RpdRedirect	ComplexTlv	25	v		
RedirectIpAddress	IpAddress	25.1	4 or 16		
ReadCount	UnsignedShort	26	2		

B.4.3 RPD Capabilities TLVs[35]

Table 13 provides the summary of GCP TLV encodings for communication of RPD Capabilities. Detailed description of the TLVs is provided in Section B.5.

Table 13 - GCP Encoding for RPD Capabilities[36]

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
RpdCapabilities	Complex TLV	50	variable	R	
NumBdirPorts	UnsignedShort	50.1	2	R	
NumDsRfPorts	UnsignedShort	50.2	2	R	
NumUsRfPorts	UnsignedShort	50.3	2	R	
NumTenGeNsPorts	UnsignedShort	50.4	2	R	
NumOneGeNsPorts	UnsignedShort	50.5	2	R	
NumDsScQamChannels	UnsignedShort	50.6	2	R	
NumDsOfdmChannels	UnsignedShort	50.7	2	R	
NumUsScQamChannels	UnsignedShort	50.8	2	R	
NumUsOfdmaChannels	UnsignedShort	50.9	2	R	
NumDsOob55d1Channels	UnsignedShort	50.10	2	R	
NumUsOob55d1Channels	UnsignedShort	50.11	2	R	
NumOob55d2Modules	UnsignedShort	50.12	2	R	
NumUsOob55d2Demodulators	UnsignedShort	50.13	2	R	
NumNdfChannels	UnsignedShort	50.14	2	R	
NumNdrChannels	UnsignedShort	50.15	2	R	
SupportsUdpEncap	Boolean	50.16	1	R	
NumDsPspFlows	UnsignedByte	50.17	1	R	Per pseudowire
NumUsPspFlows	UnsignedByte	50.18	1	R	Per pseudowire

[35] Revised per R-PHY-N-16.1451-1 on 4/15/16 and per R-PHY-N-16.1477-1 on 4/19/16 by JB.
[36] Revised per R-PHY-N-16.1564-2 on 8/22/16, per R-PHY-N-16.1573-3 on 9/8/16 and per R-PHY-N-16.1576-1 on 9/8/16 by JB. Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
RpdIdentification	Complex TLV	50.19	variable	R	
VendorName	String	50.19.1	0-255	R	
VendorId	OctetString	50.19.2	2	R	
ModelNumber	String	50.19.3	0-255	R	
DeviceMacAddress	MacAddress	50.19.4	6	R	
CurrentSwVersion	String	50.19.5	0-255	R	0-255
BootRomVersion	String	50.19.6	0-255	R	0-255
DeviceDescription	String	50.19.7	0-255	R	
DeviceAlias	String	50.19.8	0-255	R	
SerialNumber	String	50.19.9	0-255	R/W	
UsBurstReceiverVendorId	OctetString	50.19.10	2	R	
UsBurstReceiverModelNumber	String	50.19.11	3-16	R	
UsBurstReceiverDriverVersion	String	50.19.12	3-16	R	
UsBurstReceiverSerialNumber	String	50.19.13	5-16	R	
RpdRcpProtocolVersion	String	50.19.14	3-32	R	
RpdRcpSchemaVersion	String	50.19.15	5-32	R	
HwRevision	String	50.19.16	0-255	R	
AssetId	String	50.19.17	0-32	R/W	
VspSelector	String	50.19.18	0-16	R	
LcceChannelReachability	Complex TLV	50.20	variable	R	
EnetPortIndex	UnsignedByte	50.20.1	1	R	
ChannelType	RphyChannelType	50.20.2	1	R	
RfPortIndex	UnsignedByte	50.20.3	1	R	
StartChannelIndex	UnsignedByte	50.20.4	1	R	
EndChannelIndex	UnsignedByte	50.20.5	1	R	
PilotToneCapabilities	Complex TLV	50.21	variable	R	
NumCwToneGens	UnsignedByte	50.21.1	1	R	
LowestCwToneFreq	UnsignedInt	50.21.2	4	R	
HighestCwToneFreq	UnsignedInt	50.21.3	4	R	
MaxPower	TenthdBmV	50.21.4	2	R	
QamAsPilot	Boolean	50.21.5	1	R	
AllocDsChanResources	Complex TLV	50.22	variable	R	
DsPortIndex	UnsignedByte	50.22.1	1	R	
AllocatedDsOfdmChannels	UnsignedShort	50.22.2	2	R	
AllocatedDsScQamChannels	UnsignedShort	50.22.3	2	R	
AllocatedDsOob55d1Channels	UnsignedShort	50.22.4	2	R	
Deprecated	UnsignedShort	50.22.5	2	R	
AllocatedNdfChannels	UnsignedShort	50.22.6	2	R	
AllocUsChanResources	Complex TLV	50.23	variable	R	
UsPortIndex	UnsignedByte	50.23.1	1	R	
AllocatedUsOfdmaChannels	UnsignedShort	50.23.2	2	R	
AllocatedUsScQamChannels	UnsignedShort	50.23.3	2	R	
AllocatedUsOob55d1Channels	UnsignedShort	50.23.4	2	R	
Deprecated	UnsignedShort	50.23.5	2	R	

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
AllocatedNdrChannels	UnsignedShort	50.23.6	2	R	
DeviceLocation	Complex TLV	50.24	variable	R	
DeviceLocationDescription	String	50.24.1	1-255	R/W	
GeoLocationLatitude	String	50.24.2	9	R/W	
GeoLocationLogitude	String	50.24.3	10	R/W	
NumAsyncVideoChannels	UnignedByte	50.25	1	R	Per DS RF Port
SupportsFlowTags	Boolean	50.26	1	R	
SupportsFrequencyTilt	Boolean	50.27	1	R	
TiltRange	UnsignedShort	50.28	2	R	
BufferDepthMonitorAlertSupport	UnsignedByte	50.29	1	R	
BufferDepthConfigurationSupport	UnsignedByte	50.30	1	R	
RdtiCapabilies	ComplexTLV	50.30	variable	N/A	
NumPtpPortsPerEnetPort	UnsignedByte	50.30.1	1	R	
RpdUcdProcessingTime	UnsignedShort	50.31	2	R	
RpdUcdChangeNullGrantTime	UnsignedShort	50.32	2	R	
SupportMultiSectionTimingMerReporting	UnsignedByte	50.33	1	R	
RdtiCapabilities	Complex TLV	50.34	variable	R	
NumPtpPortsPerEnetPort	UnsignedByte	50.34.1	1	R	
MaxDsPspSegCount	UnsignedByte	50.35	1	R	
DirectDsFlowQueueMapping	UnsignedByte	50.36	1	R	
DsSchedulerPhbIdList	HexString	50.37	variable	R	
RpdPendingEvRepQueueSize	UnsignedShort	50.38	2	R	
RpdLocalEventLogSize	UnsignedInt	50.39	4	R	

B.4.4 RPD Operational Configuration[37]

Table 14 provides the summary of GCP TLV encodings for RPD operational configuration. Detailed description of the TLVs is provided in Section B.5.6.

Table 14 - Summary of GCP TLV Encodings used in Operational Configuration of the RPD[38]

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
EvCfg	Complex TLV	15.1	variable		
EvControl	Complex TLV	15.1.1	variable	N/A	
EvPriority	UnsignedByte	15.1.1.1	1	N/A	
EvReporting	UnsignedByte	15.1.1.2	1	R/W	
EvThrottleAdminStatus	UnsignedByte	15.1.2	1	R/W	
EvThrottleThreshold	UnsignedInt	15.1.3	4	R/W	
EvThrottleInterval	UnsignedInt	15.1.4	4	R/W	
NotifyEnable	UnsigedByte	15.1.5	1	R/W	
CcapCoreIdentification	Complex TLV	60	variable	N/A	

[37] Revised per R-PHY-N-16.1451-1 on 4/15/16 and per R-PHY-N-16.1477-1 on 4/19/16 by JB. Revised per R-PHY-N-16.1573-3 and per R-PHY-N-16.1576-1 on 9/8/16 by JB.
[38] Revised per R-PHY-N-16.1644-3 on 12/15/16 by JB.

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
Index	UnsignedByte	60.1	1	N/A	
CoreId	HexBinary	60.2	variable	R/W	
CoreIpAddress	IpAddress	60.3	4 or 16	R/W	
IsPrincipal	Boolean	60.4	1	R/W	
CoreName	String	60.5	variable	R/W	
VendorId	UnsignedShort	60.6	2	R/W	
DsRfPort	Complex TLV	61	variable		
AdminState	AdminStateType	61.2	1	R/W	
BasePower	TenthdBmV	61.3	2	R/W	TenthdBmV
RfMute	Boolean	61.4	1	R/W	
TiltSlope	UnsignedByte	61.5	1	R/W	TenthdB
TiltMaximumFrequency	UnsignedInt	61.6	4	R/W	
DedicatedToneConfig	Complex TLV	61.7	Variable		
ToneIndex	UnsignedByte	61.7.1	1	N/A	
ToneFrequency	UnsignedInt	61.7.2	4	R/W	
TonePower	TenthDb	61.7.3	1	R/W	
RfMute	UnsignedByte	61.7.4	1	R/W	
FrequencyFraction	UnsignedByte	61.7.5	1	R/W	
DsScQamChannelConfig	Complex TLV	62			
AdminState	AdminStateType	62.1	1	R/W	
CcapCoreOwner	MacAddress	62.2	6	R/W	
RfMute	Boolean	62.3	1	R/W	True if channel is muted
TSID	UnsignedShort	62.4	2	R/W	Optional attribute
CenterFrequency	UnsignedLong	62.5	4	R/W	
OperationaMode	Unsigned Byte	62.6	1	R/W	
Modulation	DsModulationType	62.7	1	R/W	
InterleaverDepth	DsInterleaveType	62.8	1	R/W	
Annex	DsAnnexType	62.9	1	R/W	
SyncInterval	UnisgnedByte	62.10	1	R/W	If zero, no Sync is sent
SyncMacAddress	MacAddress	62.11	6	R/W	
SymbolFrequencyDenominator	UnsignedShort	62.12	2	R/W	
SymbolFrequencyNumerator	UnsignedShort	62.13	2	R/W	Is this needed?
SymbolRateOverride	UnsignedLong	62.14	4	R/W	
SpectrumInversionEnabled	Bool	62.15	1	R/W	
PowerAdjust	TenthdB	62.16	1	R/W	TenthdB
DsOfdmChannelConfig	Complex TLV	63			
AdminState	AdminStateType	63.1	1	R/W	
CcapCoreOwner	MacAddress	63.2	6	R/W	
RfMute	Boolean	63.3	1	R/W	True if channel is muted
SubcarrierZeroFreq	UnsignedInt	63.4	4	R	

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
FirstActiveSubcarrier	UnsignedShort	63.5	2	R	
LastActiveSubcarrier	UnsignedShort	63.6	2	R	
NumActiveSubcarriers	UnsignedShort	63.7	2	R	
CyclicPrefix	DsOfdmCyclicPrefixType	63.8	1	R	
RollOffPeriod	DsOfdmWindowingType	63.9	1	R	
PlcFReq	UnsignedInt	63.10	4	R	
TimeInterleaverDepth	UnsignedByte	63.11	1	R	
SubcarrierSpacing	UnsignedByte	63.12	1	R	
DsOfdmSubcarrierType	Complex TLV	63.13			
StartSubcarrierId	UnsignedShort	63.13.1	2	N/A	
EndSubcarrierId	UnsignedShort	63.13.2	2	R	
SubcarrierUsage	SubcarrierUsageType	63.13.3	1	R	
DsOfdmProfile	Complex TLV	64			
ProfileId	UnsignedByte	64.1	1		key
DsOfdmSubcarrierModulation	Complex TLV	64.2	variable		
StartSubcarrierId	UnsignedShort	64.2.1	2	N/A	key
EndSubcarrierId	UnsignedShort	64.2.2	2	R	
Modulation	DsOfdmModulationType	64.2.3	1	R	
UsScQamChannelConfig	Complex TLV	65			
AdminState	AdminStateType	65.1	1	R/W	
CcapCoreOwner	MacAddress	65.2	6	R/W	
ChannelType	UpstreamChantype	65.3	1	R/W	
CenterFrequency	UnsignedInt	65.4	4	R	
Width	UnsignedInt	65.5	4	R	
SlotSize	UnsignedInt	65.6	4	R	
StartingMinislot	UnsignedInt	65.7	4	R	
PreambleString	HexBinary	65.8	variable	R	
TargetRxPower	Short	65.9	2	R/W	
IntervalUsageCode	Complex TLV	65.10			
Code	UnsignedByte	65.10.1	1	1..14	
DifferentialEncoding	Boolean	65.10.2	1	R	
FecErrorCorrectionT	UnsignedByte	65.10.3	1	R	
FecCodewordLength	UnsignedByte	65.10.4	1	R	
PreambleLen	UnsignedShort	65.10.5	2	R	
PreambleOffsett	UnsignedShort	65.10.6	2	R	
PreambleModType	PreambleType	65.10.7	1	R	
Scrambler	Boolean	65.10.8	1	R	
ScrambleSeed	UnsignedShort	65.10.9	2	R	
MaxBurstSize	UnsignedByte	65.10.10	1	R	

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
LasCodewordShortened	Boolean	65.10.11	1	R	
ByteInterleaverDepth	UnsignedByte	65.10.12	1	R	
ByteInterleaverBlockSize	UnsignedShort	65.10.13	2	R	
ModulationType	UnsignedByte	65.10.14	1	R	
GuardTime	UnsignedByte	65.10.15	1	R	
EqualizationCoeffEnable	Boolean	65.11	1	R/W	
IngressNoiseCancelEnable	Boolean	65.12	1	R/W	
UsOfdmaChannelConfig	Complex TLV	66			
AdminState	AdminStateType	66.1	1	R/W	
CcapCoreOwner	MacAddress	66.2	6	R/W	
SubcarrierZeroFreq	UnsignedInt	66.3	4	R	
FirstActiveSubcarrierNum	UnsignedShort	66.4	2	R	
LastActiveSubcarrierNum	UnsignedShort	66.5	2	R	
RollOffPeriod	UsOfdmaRollOffPeriodType	66.6	2	R	
CyclicPrefix	UsOfdmaCyclicPrefixType	66.7	2	R	
SubcarrierSpacing	SubcarrierSpacingType	66.8	1	R	
NumSymbolsPerFrame	UnsignedByte	66.9	1	R	
NumActiveSubcarriers	UnsignedShort	66.10	2	R	
StartingMinislot	UnsignedInt	66.11	4	R	
PreambleString	HexBinary	66.12	variable	R	Up to 196 B
TargetRxPower	UnsignedShort	66.13	2	R	TenthdBmV
EnableFlowTags	Boolean	66.14	1	R	
ScramblerSeed	UnsignedInt	66.15	4	R	23 LSBs
ConfigMultiSectionTimingMer	UnisignedInt	66.16	1	R/W	
BwReqAggrControlOfdma	Complex TLV	66.17			
MaxReqBlockEnqTimeout	UnsignedShort	66.17.1	2	R/W	
MaxReqBlockEnqNumber	UnsignedByte	66.17.2	1	R/W	
UsOfdmaInitialRangingIuc	Complex TLV	67	variable		
NumSubcarriers	UnsignedShort	67.1	2	R	
Guardband	UnsignedShort	67.2	2	R	
UsOfdmaFineRangingIuc	Complex TLV	68	variable		
NumSubcarriers	UnsignedShort	68.1	2	R	
Guardband	UnsignedShort	68.2	2	R	
UsOfdmaDataIuc	Complex TLV	69	variable		
DataIuc	UnsignedByte	69.1	1		key
StartMinislot	UnsignedByte	69.2	1		key
FirstSubcarrierId	UnsignedShort	69.3	2	R	
NumConsecutiveMinislots	UnsignedByte	69.4	1	R	
MinislotPilotPattern	UnsignedByte	69.5	1	R	
DataSymbolModulation	UsOfdmaModulationType	69.6	1	R	

Attribute/TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
UsOfdmaSubcarrierCfgState	Complex TLV	70	variable		
StartingSubcarrierId	UnsignedShort	70.1	2	R	
NumConsecutiveSubcarriers	UnsignedShort	70.2	2	R	
SubarrierUsage	UsOfdmaModulationType	70.3	1	R	
SidQos	ComplexTLV	96			
StartSid	UnsignedShort	96.1	2	N/A	
NumSids	UnsignedShort	96.2	2	N/A	
SidSfType	UnsignedByte	96.3	1	R/W	
SidUepiFlowId	UnsignedByte	96.4	1	R/W	
SidFlowTag	UnsignedInt	96.5	4	R/W	
UsRfPort	Complex TLV	98	variable		
AdminState	AdminStateType	98.1	1	R/W	
BwReqAggrControl	Complex TLV	98.2			
MaxReqBlockEnqTimeout	UnsignedShort	98.2.1	2	R/W	
MaxReqBlockEnqNumber	UnsignedByte	98.2.2	1	R/W	

B.4.5 Status and Performance Management TLVs

Table 15 shows the summary of TLVs defined for device management purposes.

Table 15 - Summary of RCP Status and Performance TLVs[39]

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
DsRfPortPerf	Complex TLV	71	variable	R	
DsScQamChannelPerf	Complex TLV	72	variable	R	
outDiscards	UnsignedInt	72.1	4	R	
outErrors	UnsignedInt	72.2	4	R	
DsOfdmChannelPerf	Complex TLV	73	variable	R	
outDiscards	UnsignedInt	73.1	4	R	
outErrors	UnsignedInt	73.2	4	R	
DsOfdmProfilePerf	Complex TLV	73.3	variable	R	
ProfileIndex	UnsignedByte	73.3.1	1		
outCodewords	UnsignedInt	73.3.2	4	R	
DsOob551lPerf	Complex TLV	74	variable	R	
outDiscards	UnsignedInt	74.1	4	R	
outErrors	UnsignedInt	74.2	4	R	
DsOob552Perf	Complex TLV	75	variable	R	
outDiscards	UnsignedInt	75.1	4	R	
outErrors	UnsignedInt	75.2	4	R	
NdfPerf	Complex TLV	76	variable	R	

[39] Revised per R-PHY-N-16.1477-1 on 4/19/16 by JB. Revised per R-PHY-N-16.1573-3 and per R-PHY-N-16.1576-1 on 9/8/16 by JB.

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
outDiscards	UnsignedInt	76.1	4	R	
outErrors	UnsignedInt	76.2	4	R	
UsRfPortPerf	Complex TLV	77	variable	R	
UsScQamChannelPerf	Complex TLV	78	variable	R	
UsScQamIuclPerf	Complex TLV	78.1	variable	R	
Usluc	UnsignedByte	78.1.1	1		
Collisions	UnsignedInt	78.1.2	4	R	
NoEnergy	UnsignedInt	78.1.3	4	R	
UsOfdmaChannelPerf	Complex TLV	79	variable	R	
UsOob551lPerf	Complex TLV	80	variable	R	
UsOob552Perf	Complex TLV	81	variable	R	
NdrPerf	Complex TLV	82	variable	R	
OutputBufferOccupancyHistory	Complex TLV	83	variable		
MaximumBufferDepth	UnsignedInt	83.1	4	R/O	
BufferDepth	UnsignedInt	83.2	4	R/W	
EnableMonitor	Boolean	83.3	1	R/W	
NormalizationFactor	UnsignedInt	83.4	4	R/W	
FirstSampleTimestamp	UnsignedInt	83.5	4	R/W	
SampledBufferOccupancy	HexBinary	83.6	1000	R/O	
OutputBufferThresholdAlert	Complex TLV	84	variable		
BufferDepthMonAlertEnable	Boolean	84.1	1	R/W	
BufferDepthMonAlertStatus	UnsignedByte	84.2	1	R/O	
AlertThreshold	UnsignedByte	84.3	1	R/W	
SmoothingFactorN	UnsignedShort	84.4	2	R/W	
LastAlertTimestamp	UnsignedInt	84.5	1	R/O	

B.4.6 Device Management TLVs

Table 16 shows the summary of TLVs defined for device management purposes.

Table 16 - Summary RCP device management TLVs

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
Ssd	Complex TLV	90	variable		
SsdServerAddress	IPv4 or IPv6	90.1	4 or 16		
SsdTransport	UnsignedByte	90.2	1		
SsdFilename	String	90.3	variable		
SsdStatus	UnsignedByte	90.4	1		
SsdControl	UnsignedByte	90.5	1		

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
SsdManufCvcChain	OctetString	90.6	variable		
SsdCosignerCvcChain	OctetString	90.7	variable		

B.4.7 SCTE 55-1 OOB Configuration TLVs[40]

Table 17 shows the summary of GCP TLVs defined for configuration of SCTE 55-1 out-of-band channels.

Table 17 - SCTE 55-1 Configuration TLVs

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
DsOob55d1	Complex TLV	91	variable		
AdminState	AdminStateType	91.1	1		
CcapCoreOwner	MacAddress	91.2	6		
RfMute	Boolean	91.3	1		
Frequency	UnsignedInt	91.4	4		
PowerAdjust	Byte	91.5	1		
UsOob55d1	Complex TLV	92	variable		
AdminState	AdminStateType	92.1	1		
CcapCoreOwner	MacAddress	92.2	6		
Frequency	UnsignedInt	92.3	4		
VarpdDeviceId	UnsignedInt	92.4	4		
VarpdRfPortId	UnsignedByte	92.5	1		
VarpdDemodId	UnsignedByte	92.6	1		

B.4.8 SCTE 55-2 OOB Configuration TLVs[41]

Table 18 shows the summary of GCP TLVs defined for configuration of SCTE 55-2 out-of-band functions.

Table 18 - SCTE 55-2 Configuration TLVs

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
Oob55d2Config	Complex TLV	93	variable		
DsCenterFrequency	UnsignedInt	93.1	4	R/W	
UsCenterFrequency	UnsignedInt	93.2	4	R/W	
CcapCoreOwner	MacAddress	93.3	6	R/W	
Oob55d2Module	Complex TLV	93.4	variable		
ModuleIndex	UnsignedByte	93.4.1	1	N/A	
ModulatorId	UnsignedByte	93.4.2			
ServiceChannelLastSlot	UnsignedShort	93.4.3	2	R/W	
DefaultRangingInterval	UnsignedByte	93.4.4	1	R/W	
DefaultRangingSlotConfiguration	UnsignedShort	93.4.5	2	R/W	

[40] Section added per R-PHY-N-16.1451-1 on 4/15/16 and per R-PHY-N-16.1562-1 on 8/10/16 by JB.
[41] Section added per R-PHY-N-16.1451-1 on 4/15/16 by JB.

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
DefaultNonRangingSlotConfiguration	UnsignedShort	93.4.6	2	R/W	
Randomizer	UnsignedByte	93.4.7	1	R/W	
DsPower	UnsignedByte	93.4.8	1	R/W	Units: TenthdB
DsPortAssociation	UnsignedByte	93.4.9	variable	RO	
Oob55d2Demod	Complex TLV	93.4.10	variable		
DemodIndex	UnsignedByte	93.4.10.1	1	N/A	
UpstreamGroupId	UnsignedByte	93.4.10.2	1	R/W	
MaxDhctDistance	UnsignedByte	93.4.10.3	1	R/W	
UsPortAssociation	UnsignedByte	93.4.10.4	1	RO	
RfMute	Boolean	93.4.11	1	RW	

B.4.9 NDF Configuration TLVs[42]

Table 19 shows the summary of GCP TLVs defined for configuration of NDF channels.

Table 19 - NDF Configuration TLVs

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
NdfConfig	Complex TLV	94	variable		
AdminState	AdminStateType	94.1	1	R/W	
CcapCoreOwner	MacAddress	94.2	6	R/W	
RfMute	Boolean	94.3	1	R/W	
CenterFrequency	UnsignedInt	94.4	4	R/W	Hz
ChannelWidth	UnsignedInt	94.5	4	R/W	Hz
PowerAdjust	Byte	94.6	1	R/W	Units: TenthdB

B.4.10 NDR Configuration TLVs[43]

Table 20 shows the summary of GCP TLVs defined for configuration of NDR channels.

Table 20 - NDR Configuration TLVs

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
NdrConfig	Complex TLV	95	variable		
AdminState	AdminStateType	95.1	1	R/W	
CcapCoreOwner	MacAddress	95.2	6	R/W	
CenterFrequency	UnsignedInt	95.3	4	R/W	Hz
ChannelWidth	UnsignedInt	95.4	4	R/W	Hz
NdrPower	Byte	95.5	1	R/W	Units: TenthdB

[42] Section added per R-PHY-N-16.1477-1 on 4/19/16 by JB.
[43] Section added per R-PHY-N-16.1477-1 on 4/19/16 by JB.

B.4.11 RDTI Configuration TLVs[44]

Table 21 shows the summary of GCP TLVs defined for configuration of RDTI objects in the RPD.

Table 21 - RDTI Configuration TLVs

TLV Name	Object Type	TLV Type	TLV Value Field Length	Constraints	Comments
RdtiConfig	Complex TLV	97	variable		
RpdRdtiMode	UnsignedByte	97.1	1	R/W	
RpdPtpDefDsDomainNumber	UnsignedByte	97.2	1	R/W	
RpdPtpDefDsPriority1	UnsignedByte	97.3	1	R/W	
RpdPtpDefDsPriority2	UnsignedByte	97.4	1	R/W	
RpdPtpDefDsLocalPriority	UnsignedByte	97.5	1	R/W	
RpdPtpProfileIdentifier	MacAddress	97.6	6	R/W	
RpdPtpProfileVersion	HexString	97.7	3	R/W	
RpdPtpPortConfig	Complex TLV	97.8	variable		
RpdEnetPortIndex	UnsignedShort	97.8.1	2		
RpdPtpPortIndex	UnsignedShort	97.8.2	2		
RpdPtpPortAdminState	AdminStateType	97.8.3	1	R/W	
RpdPtpPortClockSource	IpAddress	97.8.4	4 or 16	R/W	
RpdPtpPortClockAlternateSource	IpAddress	97.8.5	4 or 16	R/W	
RpdPtpPortClockSelectAlternateSourceFirst	Boolean	97.8.6	1	R/W	
RpdPtpPortTransportType	UnsignedByte	97.8.7	1	R/W	Ipv4, Ipv6
RpdPtpPortTransportCos	UnsignedByte	97.8.8			
RpdPtpPortTransportDscp	UnsignedByte	97.8.9	1	R/W	
RpdPtpPortDsLocalPriority	UnsignedByte	97.8.10	1	R/W	
RpdPtpPortDsLogSyncInterval	UnsignedByte	97.8.11	1	R/W	Base 2 scale
RpdPtpPortDsLogAnnounceInterval	UnsignedByte	97.8.12	1	R/W	Base 2 scale
RpdPtpPortDsLogDelayReqInterval	UnsignedByte	97.8.13	1	R/W	Base 2 scale
RpdPtpPortDsAnnounceReceiptTimeout	UnsignedByte	97.8.14	1	R/W	
RpdPtpPortUnicastContractDuration	UnsignedByte	97.8.15	1	R/W	
RpdPtpPortClockSrcGw	IpAddress	97.8.16	4 | 16	R/W	
RpdPtpPortClockAltSrcGw	IpAddress	97.8.17	4 | 16	R/W	

B.5 Remote PHY System Control Plane[45]

The top level UML diagram representing the complete remote PHY control plane is presented in Figure 29.

[44] Revised per R-PHY-N-16.1564-2 on 8/22/16 by JB.
[45] Added per R-PHY-N-15.1360-4 on 9/22/15 and per R-PHY-N-15.1403-3 on 1/11/16 by JB.

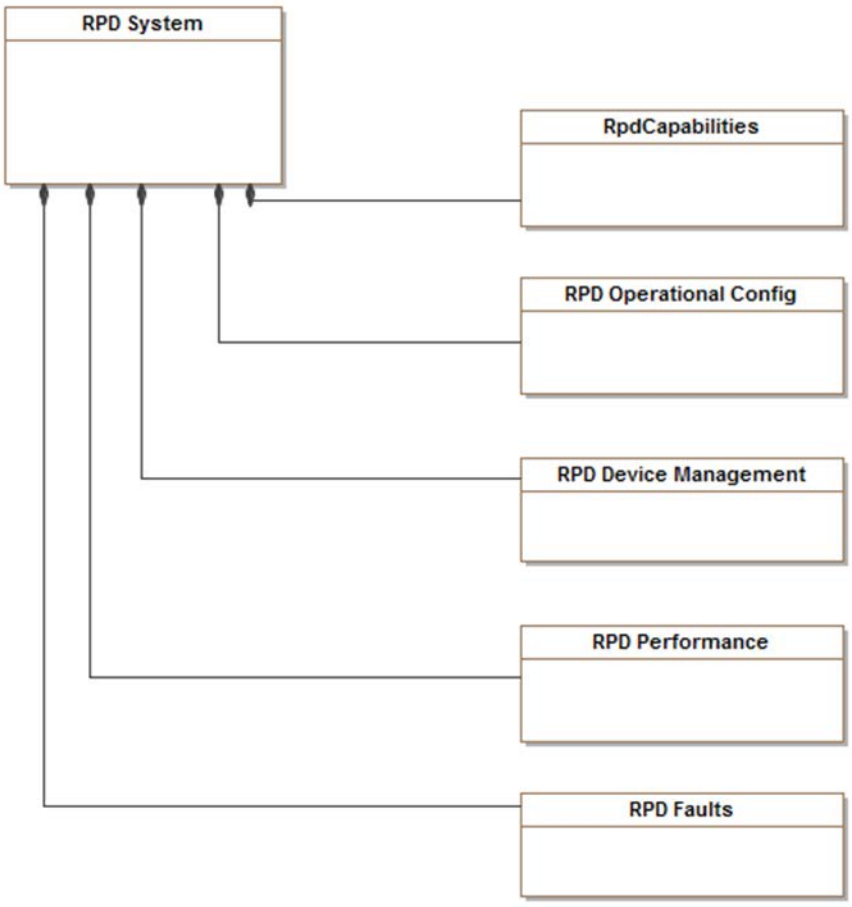

Figure 29 - RPD System Control Plane Model

B.5.1 RCP Top Level TLV

B.5.1.1 *IRA*[46]

This complex TLV represents a top level TLV encapsulating the entire RCP message and identifying the message as IRA. This TLV Value field consists of one or more Sequence (TLV 9) TLVs.

TLV Type	Length	Units	Access	Value
1	variable	N/A	N/A	One or more “Sequence” TLVs.

B.5.1.2 *REX*[47]

This complex TLV represents top level TLV encapsulating the entire RCP message and identifying the message as REX. This TLV Value field consists of one or more Sequence (TLV 9) TLVs.

TLV Type	Length	Units	Access	Value
2	variable	N/A	N/A	One or more “Sequence” TLVs.

[46] Revised per R-PHY-N-16.1673-1 on 12/15/16 by JB.
[47] Revised per R-PHY-N-16.1673-1 on 12/15/16 by JB.

B.5.1.3 *NTF*[48]

This complex TLV represents a top level TLV encapsulating the entire RCP message and identifying the message as NTF. This TLV Value field consists of one or more Sequence (TLV 9) TLVs.

TLV Type	Length	Units	Access	Value
3	variable	N/A	N/A	One or more “Sequence” TLVs.

B.5.2 RCP General Purpose TLVs

B.5.2.1 *Sequence TLV*

“Sequence” is complex TLV which represents a container for a group of RCP objects that can be exchanged via the RCP protocol. A Sequence TLV includes a single sequence number, a single operation TLV and one or more TLVs representing RCP objects.

TLV Type	Length	Access	Value
9	variable	N/A	An Operation TLV and one or more TLVs representing RCP objects on which the operation is performed.

The sender of the RCP message MUST include exactly one “Operation” TLV in the “Sequence” TLV and one or more one or more TLVs representing RCP objects.

B.5.2.2 *SequenceNumber*

The SequenceNumber TLV is used to uniquely identify sequences of RCP objects contained in the sequence TLV. This TLV is mandatory to be present in the Sequence TLV.

TLV Type	Length	Access	Value
10	2	N/A	A unique number identifying the sequence of RCP objects embedded in the “sequence” TLV. The sender of the RCP inserts the SequenceNumber value. The responder returns the same value in the response message.

The sender of RCP message MUST include the SequenceNumber TLV as the first TLV within the Sequence TLV. It is expected that the sender of RCP request message will monotonically increment the value of the SequenceNumber TLV in consecutive “Sequence” TLVs.

B.5.2.3 *Operation TLV*

The Operation TLV communicates the type of operation performed on a set of RCP objects contained with a Sequence TLV. The RCP protocol defines three operation types: “Read”, “Write”, “Delete” and corresponding types used in response messages: “ReadResponse”, “WriteResponse” and “DeleteResponse”. This TLV is mandatory to be present in the Sequence TLV.

TLV Type	Length	Access	Value
11	1	N/A	An unsigned byte representing a channel type. Valid values are: 1 - “Read”. 2 - “Write” 3 - “Delete” 4 - “ReadResponse” 5 - “WriteResponse” 6 - “DeleteResponse”

The sender of RCP message MUST include the Operation TLV as the second TLV within the Sequence TLV.

[48] Revised per R-PHY-N-16.1673-1 on 12/15/16 by JB.

B.5.2.4 *RfChannelSelector TLV*

The RfChannelSelector TLV is a complex TLV used to identify an RF channel in the RPD. The RfChannelSelector TLV value field contains three sub-TLVs defining RF Port Index, RF Channel Type and RF Channel Index.

TLV Type	Length	Access	Value
12	12	N/A	The value field includes exactly three sub-TLVs: 12.1, 12.2 and 12.3.

B.5.2.5 *RfPortIndex TLV*[49]

A TLV which value represents the index of RPDs RF Port.

TLV Type	Length	Access	Value
12.1	1	N/A	The value is an unsigned byte representing the index of the RPD's RF Port to which a channel or a sub-channel belongs. The value of this field uniquely identifies US or DS RF port in the RPD. The selection of US or DS RF port depends on TLV 12.2. The valid range for DS RF ports is from 0 to NumDsRfPorts - 1. The valid range for US RF ports is from 0 to NumUsRfPorts - 1.

B.5.2.6 *RfChannelType TLV*[50]

A TLV, the value of which represents the type of a channel.

TLV Type	Length	Access	Value
12.2	1	N/A	An unsigned byte representing a channel type. Valid values are: 1 - DsScQam, downstream QAM channel. 2 - DsOfdm, downstream OFDM channel 3 - Ndf 4 - DsScte55d1, downstream SCTE 55-1 channel 5 - UsAtdma, upstream ATDMA channel 6 - UsOfdma, upstream OFDMA 7 - reserved 8 - Ndr 9 - UsScte55d1, upstream SCTE 55-1 channel All other values are reserved.

B.5.2.7 *RfChannelIndex TLV*[51]

A TLV which value represents the index of RPDs RF Port.

TLV Type	Length	Access	Value
12.3	1	N/A	An unsigned byte representing an index of RF channel of the type selected by TLV 12.2. The valid range is from 0 to N-1, where N is the value advertised by the RPD as a capability for a number of supported channels of the selected type. For example, if the RPD advertises that it supports 128 SC-QAM channels through NumDsScQamChannels capability, than the valid value of RfChannelIndex for downstream QAM channels is from 0 to 127.

[49] Revised per R-PHY-N-16.1644-3 on 12/19/16 by JB.
[50] Revised per R-PHY-N-16.1451-1 on 4/15/16 by JB.
[51] Revised per R-PHY-N-16.1644-3 on 12/19/16 by JB.

B.5.2.8 *RfPortSelector TLV*

The RfPortSelector TLV is a complex TLV which identifies an RF Port in the RPD. The RfPortSelector TLV value field contains two sub-TLVs defining RF Port Index and RF Port Type.

TLV Type	Length	Access	Value
13	8	N/A	The value field includes exactly two sub-TLVs: 13.1 and 13.2.

B.5.2.9 *RfPortIndex1 TLV*[52]

A TLV which value represents the index of RPDs RF Port.

TLV Type	Length	Access	Value
13.1	1	N/A	The value is an unsigned byte representing the index of the selected RPD's RF Port. The value uniquely identifies US or DS RF port in the RPD. The selection of US or DS RF port depends on TLV 13.2. The valid range for DS RF ports is from 0 to NumDsRfPorts - 1. The valid range for US RF ports is from 0 to NumUsRfPorts - 1.

B.5.2.10 *RfPortType TLV*

A TLV which value represents the index of RPDs RF Port.

TLV Type	Length	Access	Value
13.2	1	N/A	An unsigned byte representing a RF Port type. Valid values are: 1 - DsRfPort, downstream port. 2 - UsRfPort, upstream port All other values are reserved. The RCP currently does not define management objects for the Bi-directional RF Port.

B.5.2.11 *EnetPortIndex TLV*

The EnetPortIndex TLV is used to select an Ethernet Port in the RPD. The EnetPortIndex TLV value field contains an index uniquely identifying an Ethernet port in the RPD.

TLV Type	Length	Access	Value
14	1	N/A	An unsigned byte representing the index of the RPD's Ethernet Port.

B.5.2.12 *RpdGlobal TLV*

The RpdGlobal TLV is used as a container for a group of objects applicable to the entire RPD.

TLV Type	Length	Access	Value
15	variable	N/A	A set of TLVs consisting of global objects associated with the RPD.

B.5.2.13 *RfChannel TLV*

The RfChannel TLV is used as a container for a group of objects related to a single RF channel.

[52] Revised per R-PHY-N-16.1644-3 on 12/19/16 by JB.

TLV Type	Length	Access	Value
16	variable	N/A	A set of TLVs consisting of a single RfChannelSelector and one or more TLV representing objects associated with this channel.

B.5.2.14 *RfPort TLV*

The RfPort TLV is used as a container for a group of objects related to a single RF Port.

TLV Type	Length	Access	Value
17	variable	N/A	A set of TLVs consisting of a single RfPortSelector and one or more TLV representing objects associated with this RF port.

B.5.2.15 *EnetPort TLV*

The EnetPort TLV is used as a container for a group of objects related to a single Ethernet Port.

TLV Type	Length	Access	Value
18	variable	N/A	A set of TLVs consisting of a single EnetPortIndex and one or more TLV representing objects associated with this Ethernet Port.

B.5.2.16 *ResponseCode TLV*

The ResponseCode TLV is used to communicate an error code during processing of a request.

TLV Type	Length	Access	Value
19	1	N/A	An unsigned byte value signifying an error in processing of a request. The valid values are: 0 - OK 1 - General Error Additional values will be defined in future revisions of this specification.

B.5.2.17 *ErrorMessage TLV*

The ErrorMessage TLV is used by the RPD to communicate a string describing the error with processing of a request. The content of error messages is RPD vendor specific. This specification does not define the format or the content of error messages communicated via this TLV.

TLV Type	Length	Access	Value
20	1-255	N/A	A string with RPD vendor specific message describing the error with processing of a request.

CCAP Core MUST log Response Codes and associated Error Messages.

B.5.2.18 *VendorSpecificExtension TLV*

The VendorSpecificExtension TLV is used to communicate information vendor-specific information exchanged via RCP. The sender of the VendorSpecificExtension TLV must include a single VendorId TLV as the first sub-TLV of this TLV. The definition of additional sub-TLV is outside of the scope of this specification. Additional rules for exchange of vendor specific information are defined in section B.3.12

TLV Type	Length	Access	Value
21	7-65000	W	Two or more sub-TLVs identifying the vendor and providing vendor-specific information. Only the VendorId sub-TLV is defined in this specification. The definition of other sub-TLVs are left to vendors'' documentation.

B.5.2.19 *VendorId TLV*

This TLV communicates vendor id of the manufacturer defining vendor-specific extension as the IANA-assigned "SMI Network Management Private Enterprise Codes" [RFC 1700] value.

TLV Type	Length	Access	Value
21.1	2	N/A	An unsigned short with Vendor Id of manufacturer defining vendor-specific information.

B.5.2.20 *DocsisMsg TLV*

This TLV communicates the content of a DOCSIS message from the CCAP Core to the RPD. The rules for used of this TLV are defined in section B.3.13

TLV Type	Length	Access	Value
22	variable	N/A	An octet string containing the entire DOCSIS message starting from the DOCSIS header and ending with a CRC. The CRC value does not need to be valid.

B.5.2.21 *DocsisTimestamp32*

This TLV communicates the value of 32-bit DOCSIS Timestamp.

TLV Type	Length	Access	Value
23	4	N/A	An unsigned integer containing 32-bit DOCSIS timestamp.

B.5.2.22 *DocsisTimestamp64*

This TLV communicates the value of extended 64-bit DOCSIS Timestamp.

TLV Type	Length	Access	Value
24	8	N/A	An unsigned long containing the extended 64-bit DOCSIS timestamp.

B.5.2.23 *RpdRedirect*

This TLV is used to communicate an ordered list of CCAP Cores that the RPD is redirected to.

TLV Type	Length	Access	Value
25	variable	N/A	An ordered list of IP addresses of CCAP Cores.

B.5.2.24 *RpdRedirectIpAddress*

This TLV communicates an Ipv4 address of CCAP Core that the RPD is redirected to.

TLV Type	Length	Access	Value
25.1	4 or 16	N/A	An IPv4 or Ipv6 address of CCAP Core. The length signifies whether the address is Ipv4 or Ipv6.

B.5.3 RPD Capabilities and Identification[53]

Immediately after authentication the CCAP CCAP Core reads a set of parameters identifying the RPD, its capabilities and its available resources via the IRA message. The set of RCP capabilities and identification objects is grouped intoRPD Capabilities diagram which is presented in Figure 30.

[53] Added per R-PHY-N-15.1360-4 on 9/22/15 by JB.

The CCAP Core can also read Capabilities and Identification objects after initialization via the REX message.

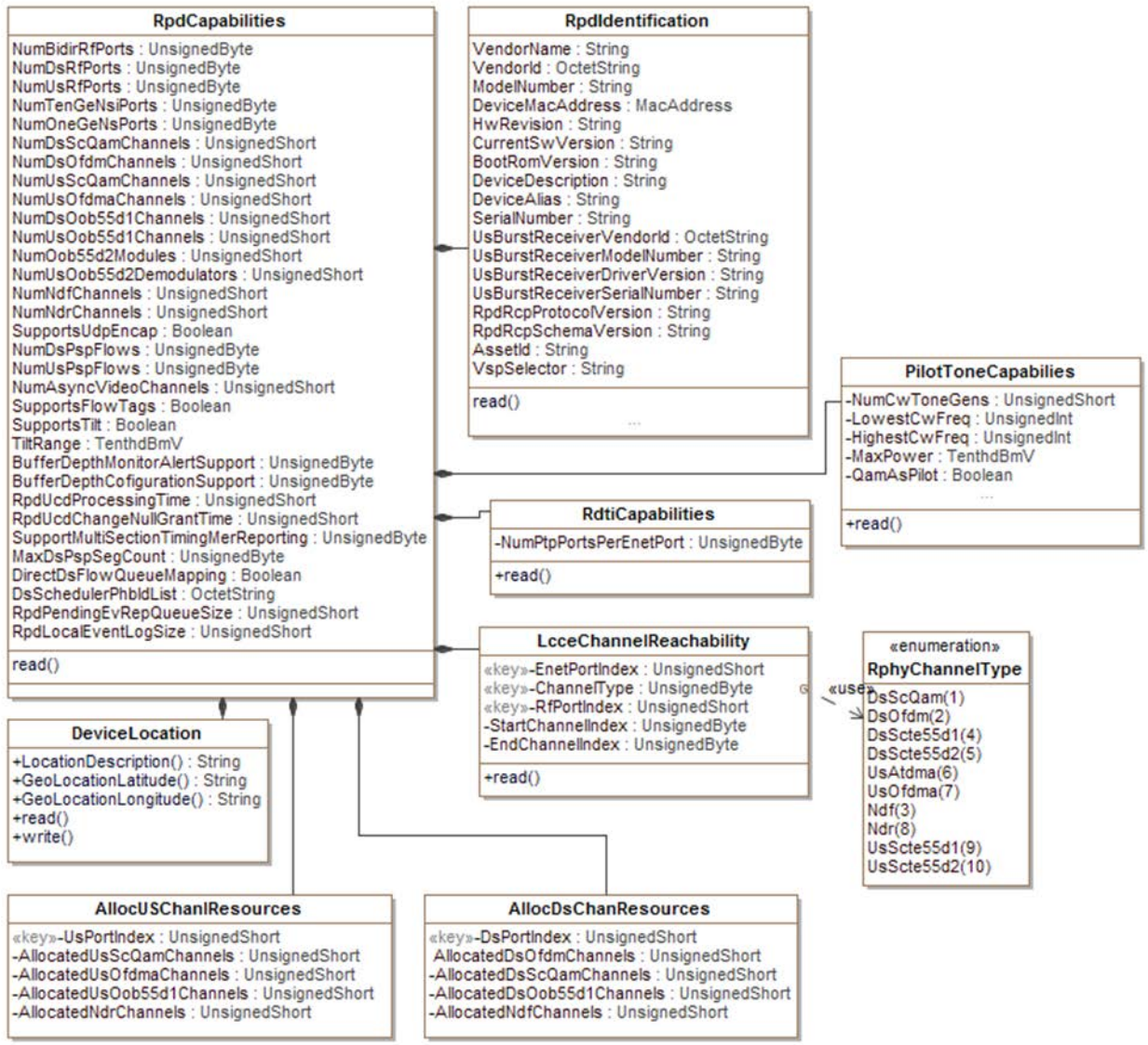

Figure 30 - RPD Capabilities Objects [54]

B.5.3.1 *RPD Capabilities* [55]

The RPD Capabilities (RpdCapabilities object) groups the fundamental capabilities of the RPD such as the supported counts of RF and Ethernet Ports and various channel counts per port.

B.5.3.1.1 NumBdirPorts

This object represents the total number of bi-direction RF ports supported by the RPD.

[54] Revised figure per R-PHY-N-16.1451-1 on 4/15/16 by JB. Revised per R-PHY-N-16.1644-3 on 12/19/16 by JB.
[55] Revised per R-PHY-N-16.1644-3 on 12/19/16 by JB.

TLV Type	Length	Units	Access	Value
50.1	2		R	An unsigned short reporting the total number of RF bi-directional ports on the RPD.

B.5.3.1.2 NumDsRfPorts

This object represents the total number of downstream RF ports supported by the RPD.

TLV Type	Length	Units	Access	Value
50.2	2		R	An unsigned short reporting the total number of downstream RF ports on the RPD.

B.5.3.1.3 NumUsRfPorts

This object represents the total number of upstream RF ports supported by the RPD.

TLV Type	Length	Units	Access	Value
50.3	2		R	An unsigned short reporting the total number of upstream RF ports on the RPD.

B.5.3.1.4 NumTenGeNsPorts

This object represents the total number of 10 Gigabit Ethernet ports supported by the RPD.

TLV Type	Length	Units	Access	Value
50.4	2		R	An unsigned short reporting the total number of 10 Gigabit Ethernet ports supported by the RPD.

B.5.3.1.5 NumOneGeNsPorts

This object represents the total number of 1 Gigabit Ethernet ports supported by the RPD.

TLV Type	Length	Units	Access	Value
50.5	2		R	An unsigned short reporting the total number of 1 Gigabit Ethernet ports supported by the RPD.

B.5.3.1.6 NumDsScQamChannels

This object represents the number of downstream SC-QAM channels per DS RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.6	2		R	An unsigned short reporting the number of downstream SC-QAM channels per DS RF port supported by the RPD.

B.5.3.1.7 NumDsOfdmChannels

This object represents the number of downstream OFDM channels per DS RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.7	2		R	An unsigned short reporting the number of downstream OFDM channels per DS RF port supported by the RPD.

B.5.3.1.8 NumUsScQamChannels

This object represents the number of upstream SC-QAM channels per US RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.8	2		R	An unsigned short reporting the number of upstream SC-QAM channels per US RF port supported by the RPD.

B.5.3.1.9 NumUsOfdmaChannels

This object represents the number of upstream OFDMA channels per US RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.9	2		R	An unsigned short reporting the number of upstream OFDMA channels per US RF port supported by the RPD.

B.5.3.1.10 NumDsOob55d1Channels

This object represents the number of downstream SCTE-55-1 channels per DS RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.10	2		R	An unsigned short reporting the number of downstream SCTE-55-1 channels per DS RF port supported by the RPD.

B.5.3.1.11 NumUsOob55d1Channels[56]

This object represents the number of upstream SCTE-55-1 channels per US RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.11	2		R	An unsigned short reporting the number of upstream SCTE-55-1 channels per US RF port supported by the RPD.

B.5.3.1.12 NumOob55d2Modules[57]

This object represents the number of SCTE-55-2 modules supported by the RPD.

TLV Type	Length	Units	Access	Value
50.12	2		R	An unsigned short reporting the number of SCTE 55-2 modules supported by the RPD.

[56]. Revised per R-PHY-N-16.1451-1 on 4/15/16 by JB
[57] Revised per R-PHY-N-16.1451-1 on 4/15/16 by JB

B.5.3.1.13 NumUsOob55d2Demodulators[58]

This object represents the number of upstream demodulators per SCTE 55-2 module supported by the RPD.

TLV Type	Length	Units	Access	Value
50.13	2		R	An unsigned short reporting the number of upstream demodulators per SCTE 55-2 module supported by the RPD.

B.5.3.1.14 NumNdfChannels

This object represents the number of Narrowband Digital Forward (NDF) channels per DS RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.14	2		R	An unsigned short reporting the number NDF of channels per DS RF port supported by the RPD.

B.5.3.1.15 NumNdrChannels

This object represents the number of Narrowband Digital Return (NDR) channels per US RF port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.15	2		R	An unsigned short reporting the number of Narrowband Digital Return (NDR) channels per US RF port supported by the RPD.

B.5.3.1.16 SupportsUdpEncap

This object allows the RPD to indicate whether it supports UDP encapsulation on L2TPv3 pseudowires.

TLV Type	Length	Units	Access	Value
50.16	1		R	A Boolean value indication whether the RPD supports UDP encapsulation. 0 - The RPD does not support UDP encapsulation on L2TPv3 pseudowires. 1 - The RPD supports UDP encapsulation on L2TPv3 pseudowires.

B.5.3.1.17 NumDsPspFlows

This object represents the number of distinct PSP Flows supported by the RPD on downstream data pseudowires.

TLV Type	Length	Units	Access	Value
50.17	1		R	An unsigned byte reporting the number of distinct PSP Flows supported by the RPD on downstream data pseudowires.

B.5.3.1.18 NumUsPspFlows

This object represents the number of distinct PSP Flows supported by the RPD on upstream data pseudowires.

[58] Revised per R-PHY-N-16.1451-1 on 4/15/16 by JB

TLV Type	Length	Units	Access	Value
50.18	1		R	An unsigned byte reporting the number of distinct PSP Flows supported by the RPD on upstream data pseudowires.

B.5.3.2 *RPD Identification TLV*

B.5.3.2.1 RpdIdentification

A complex TLV through which the RPD communicates a set of identifying parameters.

TLV Type	Length	Units	Access	Value
50.19	Variable		R	A set sub-TLV elements defined below.

B.5.3.2.2 VendorName

The VendorName object identifies the RPD's manufacturer. The detailed format is vendor proprietary.

TLV Type	Length	Units	Access	Value
50.19.1	0-255		R	A string identifying the RPD's manufacturer.

B.5.3.2.3 VendorId

This TLV communicates the RPD's manufacturer's vendor id as the IANA-assigned "SMI Network Management Private Enterprise Codes" [RFC 1700] value.

TLV Type	Length	Units	Access	Value
50.19.2	2		R	An unsigned short with Vendor Id of the RPD's manufacturer.

B.5.3.2.4 ModelNumber

This TLV convey the model name and number assigned to the RPD. The format of the string is vendor proprietary.

TLV Type	Length	Units	Access	Value
50.19.3	0-255		R	A string identifying the RPD's model number.

B.5.3.2.5 DeviceMacAddress

This TLV convey the main MAC address of the RPD. Typically the MAC address associated with the lowest numbered CIN facing Ethernet port.

TLV Type	Length	Units	Access	Value
50.19.4	6		R	The MAC address used to uniquely identify the RPD.

B.5.3.2.6 CurrentSwVersion

This TLV conveys the SW version currently running on of the RPD. The format of the string is vendor proprietary.

TLV Type	Length	Units	Access	Value
50.19.5	0-255		R	A string representing the SW version currently running on of the RPD.

B.5.3.2.7 BootRomVersion

This TLV conveys the version of the BootRom currently installed on of the RPD. The format of the string is vendor proprietary.

TLV Type	Length	Units	Access	Value
50.19.6	0-255		R	A string representing the BootRom version currently installed on of the RPD.

B.5.3.2.8 DeviceDescription

This TLV conveys a short description of the RPD in the form a string, selected by the RPD's manufacturer.

TLV Type	Length	Units	Access	Value
50.19.7	0-255		R	A string selected by the RPD manufacturer.

B.5.3.2.9 DeviceAlias

This TLV communicates a device name assigned by the operator via management interface. This object is an 'alias' name for the device as specified by a network manager, and provides a non-volatile 'handle' for the RPD.The CCAP Core MAY configure the DeviceAlias attribute per operator's request. When this attribute is written, the RPD stores its value in its non-volatile configuration.

TLV Type	Length	Units	Access	Value
50.19.8	0-255		R/W	A string communicating device's name assigned by the operator.

B.5.3.2.10 SerialNumber

This TLV communicates RPD's serial number. The format of the string is vendor proprietary.

TLV Type	Length	Units	Access	Value
50.19.9	0-16		R	A string representing device's serial number.

B.5.3.2.11 UsBurstReceiverVendorId

This TLV is used to communicate the identifier of the manufacturer of RPD's US burst receiver as the IANA-assigned value as defined in "SMI Network Management Private Enterprise Codes" [RFC 1700].

TLV Type	Length	Units	Access	Value
50.19.10	2	N/A	R	An octet string representing the identifier of the manufacturer of RPD's US burst receiver.

B.5.3.2.12 UsBurstReceiverModelNumber

This TLV is used to communicate the model number identifying RPD's US burst receiver.

TLV Type	Length	Units	Access	Value
50.19.11	3-16	N/A	R	A string with the identifier of the model number of the RPD's US burst receiver.

B.5.3.2.13 UsBurstReceiverDriverVersion

This TLV is used to communicate the version of the driver of RPD's US burst receiver.

TLV Type	Length	Units	Access	Value
50.19.12	3-16	N/A	R	A string identifying the version of the driver of the RPD's US burst receiver.

B.5.3.2.14 UsBurstReceiverSerialNumber

This TLV is used to communicate the serial number of RPD's US burst receiver.

TLV Type	Length	Units	Access	Value
50.19.13	5-16	N/A	R	A string identifying the serial number of the RPD's US burst receiver.

B.5.3.2.15 RpdRcpProtocolVersion

This TLV is used to communicate the version of the RCP protocol supported by the RPD.

TLV Type	Length	Units	Access	Value
50.19.14	3-32	N/A	R	A string identifying the RCP protocol version supported by the RPD.

The RPD MUST report the RPC protocol version as "1.0". In future versions of this specification, this TLV will be used in protocol version negotiation.

B.5.3.2.16 RpdRcpSchemaVersion

This TLV is used to communicate the version of the RCP schema supported by the RPD.

TLV Type	Length	Units	Access	Value
50.19.15	5-32	N/A	R	A string identifying the RCP schema version supported by the RPD.

The RPD MUST report the RPC schema protocol version as "1.0.X", where X is a number selected by the RPD manufacturer. In future versions of this specification this TLV will be used in schema and protocol version negotiation.

B.5.3.2.17 HwRevision

This TLV is used to communicate the revision of the RPD hardware.

TLV Type	Length	Units	Access	Value
50.19.16	0-255	N/A	R	A string identifying the revision of the RPD hardware.

B.5.3.2.18 AssetId[59]

This attribute is modeled after entPhysicalAssetID object defined in RFC 6933. AssetId is used to communicate the asset tracking identifier as assigned by a network manager. When this attribute is written, the RPD stores its value in its non-volatile configuration.

TLV Type	Length	Units	Access	Value
50.19.17	0-32	N/A	R/W	A string containing asset identification of the RPD. The default value is the zero-length string or "".

B.5.3.2.19 VspSelector[60]

The RPD advertises VspSelector (VSP stands for Vendor-Specific Pre-configuration) in the form of human readable string. VspSelector is used by the Principal CCAP Core to match the RPD to VSP configuration maintained on the CCAP Core and to deliver VSP configuration to the RPD during initialization. The details of VSP operation are explained in Section B.2.12.1. The VSP configuration maintained on the CCAP Core is described in [R-OSSI]

TLV Type	Length	Units	Access	Value
50.19.18	0-16	N/A	R	A string containing a VSP Selector. If the RPD does not support VSP the RPD communicates VSP as a zero-length string.

[59] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[60] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

B.5.3.3 *LCCE Channel Reachability*

B.5.3.3.1 LcceChannelReachability

This object permits the RPD to report connectivity constrains between the CIN facing Ethernet ports and the channels supported on RF ports of the RPD.

TLV Type	Length	Units	Access	Value
50.20	variable		R	A set of sub-TLV elements defined below.

B.5.3.3.2 EnetPortIndex

This object uniquely identifies an Ethernet port on the RPD.

TLV Type	Length	Units	Access	Value
50.20.1	1		N/A	An unsigned byte representing an RPD's Ethernet port index.

B.5.3.3.3 ChannelType

This object represent a channel type.

TLV Type	Length	Units	Access	Value
50.20.2	1		N/A	An unsigned byte representing a channel type. Valid values are: 1 - DsScQam, downstream QAM channel. 2 - DsOfdm, downstream OFDM channel 3 - Ndf 4 - DsScte55d1, downstream SCTE 55-1 channel 5 - UsAtdma, upstream ATDMA channel 6 - UsOfdma, upstream OFDMA 7 - Reserved 8 - Ndr 9 - UsScte55d1, upstream SCTE 55-1 channel All other values are reserved.

B.5.3.3.4 RfPortIndex[61]

This object identifies the RPD's RF port. This object can represent an upstream or a downstream port depending on the value of ChannelType TLV.

TLV Type	Length	Units	Access	Value
50.20.3	1		N/A	An unsigned byte identifying an RF port of the RPD. The valid range is defined by RPD capabilities, The valid range for DS RF ports is from 0 to NumDsRfPorts - 1. The valid range for US RF ports is from 0 to NumUsRfPorts - 1.

B.5.3.3.5 StartChannelIndex[62]

This object identifies the first channel in the reported connectivity range.

[61] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[62] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
50.20.4	1		R	An unsigned byte representing the first channel in reported range. The valid range is from 0 to N-1, where N is the value advertised by the RPD as a capability for a number of supported channels of the selected type.

B.5.3.3.6 EndChannelIndex[63]

This object identifies the last channel in the reported connectivity range.

TLV Type	Length	Units	Access	Value
50.20.5	1		R	An unsigned byte representing the last channel in reported range The valid range is from 0 to N-1, where N is the value advertised by the RPD as a capability for a number of supported channels of the selected type.

B.5.3.4 *PilotToneCapabilities*

The RPD communicates its ability to generate CW carriers that can serve as pilot tones and alignment carriers through this object.

TLV Type	Length	Units	Access	Value
50.21	variable		R	A set of sub-TLV elements defined below.

B.5.3.4.1 NumCwToneGens

This object allows the RPD to convey the number of supported CW carrier generators per DS RF port.

TLV Type	Length	Units	Access	Value
50.21.1	1		R	An unsigned byte reporting the number of dedicated CW tone generators per DF RF Port supported by the RPD.

B.5.3.4.2 LowestCwToneFreq

This object permits the RPD to inform the CCAP Core about the lowest frequency supported by the dedicated CW tone generators.

TLV Type	Length	Units	Access	Value
50.21.2	4		R	An unsigned integer reporting the lowest frequency supported by the dedicated CW tone generators.

B.5.3.4.3 HighestCwToneFreq

This object permits the RPD to inform the CCAP Core about the highest frequency supported by the dedicated CW tone generators.

TLV Type	Length	Units	Access	Value
50.21.3	4		R	An unsigned integer reporting the highest frequency supported by the dedicated CW tone generators.

B.5.3.4.4 MaxPower

The object allows the RPD to inform the CCAP Core what is the maximum power level supported by the dedicated CW tone generators.

[63] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
50.21.4	2	TenthdBmV	R	An unsigned short reporting the maximum power level of a dedicated CW tone supported by the RPD.

B.5.3.4.5 QamAsPilot

Through this object the RPD informs the CCAP Core whether its QAM channels can be configured as CW tones.

TLV Type	Length	Units	Access	Value
50.21.5	1		R	A Boolean value indicating whether the RPD support configuration of QAM channels as CW tones. Supported values are: 0 - The RPD does not support configuration of QAM channels as CW tones. 1 - The RPS supports configuration of QAM channels as CW tones. All other values are reserved.

B.5.3.5 *Allocated Downstream Channel Resources*

B.5.3.5.1 AllocDsChanResources

The RPD reports the allocation status for its downstream channel resources through AllocDsChanResources this complex TLV.

TLV Type	Length	Units	Access	Value
50.22	variable		R	A set of sub-TLV elements defined below.

B.5.3.5.2 DsPortIndex

This object specifies a unique index for the downstream RF port.

TLV Type	Length	Units	Access	Value
50.22.1	1		R	An unsigned byte with a zero based index identifying RPDs downstream RF Port. The valid range is from 0 to NumDsRfPorts - 1.

B.5.3.5.3 AllocatedDsOfdmChannels

The object allows the RPD to inform the CCAP Core how many DS OFDM channels have been allocated on the selected RF port of the RPD.

TLV Type	Length	Units	Access	Value
50.22.2	2		R	An unsigned short which indicates how many DS OFDM channels have been allocated on the selected RF port of the RPD.

B.5.3.5.4 AllocatedDsScQamChannels

The object allows the RPD to inform the CCAP Core how many DS QAM channels have been allocated on the selected RF port of the RPD.

TLV Type	Length	Units	Access	Value
50.22.3	2		R	An unsigned short which indicates how many DS QAM channels have been already allocated on the selected RF port of the RPD.

B.5.3.5.5 AllocatedDsOob55d1Channels

The object allows the RPD to inform the CCAP Core how many DS out-of-band SCTE-55-1 channels have been allocated on the selected RPD's RF port.

TLV Type	Length	Units	Access	Value
50.22.4	2		R	An unsigned short which indicates how many DS out-of-band SCTE-55-1 channels have been allocated on the selected RF port of the RPD.

B.5.3.5.6 AllocatedNdfChannels

The object allows the RPD to inform the CCAP Core how many DS NDF channels have been allocated on the selected RPD's RF port.

TLV Type	Length	Units	Access	Value
50.22.6	2		R	An unsigned short indicating how many DS NDF channels have been allocated on the selected RF port of the RPD.

B.5.3.6 *Allocated Upstream Channel Resource*

B.5.3.6.1 AllocUsChanResources

The RPD reports the allocation status for its upstream channel resources through AllocUsChanResources object.

TLV Type	Length	Units	Access	Value
50.23	variable		R	A set of sub-TLV elements defined below.

B.5.3.6.2 UsPortIndex

This object specifies a unique index for the upstream RF port.

TLV Type	Length	Units	Access	Value
50.23.1	1		N/A	An unsigned byte with a zero based index identifying RPDs upstream RF Port. The valid range is from 0 to NumUsRfPorts - 1.

B.5.3.6.3 AllocatedUsOfdmaChannels

The object allows the RPD to inform the CCAP Core how many US OFDMA channels have been already allocated on the selected RF port of the RPD.

TLV Type	Length	Units	Access	Value
50.23.2	2		R	An unsigned short which indicates how many US OFDM channels have been already allocated on the selected RF port of the RPD.

B.5.3.6.4 AllocatedUsScQamChannels

The object allows the RPD to inform the CCAP Core how many US SC-QAM channels have been allocated on the selected RF port of the RPD.

TLV Type	Length	Units	Access	Value
50.23.3	2		R	An unsigned short which indicates how many US SC-QAM (ATDMA) channels have been already allocated on the selected RF port of the RPD.

B.5.3.6.5 AllocatedUsOob55d1Channels

The object allows the RPD to inform the CCAP Core how many upstream out-of-band SCTE-55-1 channels have been already allocated on the selected RPD's RF port.

TLV Type	Length	Units	Access	Value
50.23.4	2		R	An unsigned short which indicates how many US SCTE 55-1 out-of-band channels have been already allocated on the selected RF port of the RPD.

B.5.3.6.6 AllocatedNdrChannels

The object allows the RPD to inform the CCAP Core how many NDR channels have been already allocated on the selected RF port of the RPD.

TLV Type	Length	Units	Access	Value
50.23.6	1		R	An unsigned short which indicates how many NDR channels have been already allocated on the selected RF port of the RPD.

B.5.3.7 *Device Location*[64]

This TLV allows the RPD to inform the CCAP Core about it its location. The location information is configured during the RPD installation. The RPD retains the location information in its non-volatile memory.

TLV Type	Length	Units	Access	Value
50.24	variable		R	A complex TLV grouping TLVs used to convey RPD's location information.

B.5.3.7.1 Device Location Description

This object allows the RPD to inform the CCAP Core about it its location. The format of the information is specific to a cable operator. The CCAP Core MAY configure the DeviceLocationDescription attribute per operator's request. When this attribute is written, the RPD stores its value in its non-volatile configuration.

TLV Type	Length	Units	Access	Value
50.24.1	1-255		R/W	A string with a short text description of where the RPD has been installed, such as a street address. The format is specific to the operator.

B.5.3.7.2 GeoLocationLatitude

This object allows the RPD to inform the CCAP Core about the latitude portion of its geographic location. The CCAP Core MAY configure the GeoLocationLatitude attribute per operator's request. When this attribute is written, the RPD stores its value in its non-volatile configuration.

TLV Type	Length	Units	Access	Value
50.24.2	9		R	A 9 byte long string with RPD's latitude formatted as in ISO 6709-2008. The RPD uses "6 digit notation" in the format deg, min, sec, ±DDMMSS.S. example: -750015.1

[64] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

B.5.3.7.3 DeviceGeoLocationLongitude[65]

This object allows the RPD to inform the CCAP Core about the longitude portion of its geographic location.The CCAP Core MAY configure the DeviceGeoLocationLongitude attribute per operator's request. When this attribute is written, the RPD stores its value in its non-volatile configuration.

TLV Type	Length	Units	Access	Value
50.24.3	10		R	A 10 byte long string with RPD's latitude formatted as in ISO 6709-2008. The RPD uses "7 digit notation" in the format deg, min, sec, ±DDDMMSS.S. example: -0100015.1

B.5.3.7.4 NumAsyncVideoChannels [66]

This object represents the number of asynchronous MPEG video channels per DF RF Port supported by the RPD.

TLV Type	Length	Units	Access	Value
50.25	1		R	An unsigned byte reporting the number of asynchronous MPEG video channels per DS RF Port supported by the RPD.

B.5.3.7.5 SupportsFlowTags[67]

This TLV reports Flow Tags support capability. If the value is set to 1, the RPD supports Flow Tags on OFDMA channels. A Flow Tag is a 32-bit identifier of a MAC hardware resource (typically a Service Flow). The Flow Tag can be assigned to the scheduled SID by the CCAP Core. The RPD provides Flow Tags in UEPI headers for OFDMA channels.

TLV Type	Length	Units	Access	Value
50.26	1		R	A Boolean value indication whether the RPD supports Flow Tags for OFDMA channels. 0 - The RPD does not support Flow Tags. 1 - The RPD supports FlowTags.

B.5.3.7.6 SupportsFrequencyTilt[68]

This TLV reports Frequency Tilt support capability. If the value is set to 1, the RPD supports Frequency Tilt settings on DS RF Ports.

TLV Type	Length	Units	Access	Value
50.27	1		R	A Boolean value indication whether the RPD supports Frequency Tilt settings on DS RF port. 0 - The RPD does not support Frequency Tilt settings. 1 - The RPD supports Frequency Tilt settings.

B.5.3.7.7 TiltRange[69]

This TLV reports the range of tilt setting that the RPD supports.

[65] Revised per R-PHY-N-16.1451-1 on 4/15/16 by JB.
[66] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[67] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[68] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[69] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
50.28	1	TenthdB	R	An unsigned byte value specifying the maximum tilt slope setting that can be supported by the RPD in units of 0.1 dB.

B.5.3.7.8 BufferDepthMonitorAlertSupport[70]

The RPD communicates its capability to support for buffer depth monitoring Alerts through this TLV.

TLV Type	Length	Units	Access	Value
50.29	1	N/A	R	An unsigned byte, a bitmask specifying the capability of the RPD to support Buffer Depth Monitoring Alerts. If set to '1' the bits listed below indicate the capability to monitor buffer depth on the corresponding types of downstream channels. Bit 0 - OFDM channels Bit 1 - SC-QAM DOCSIS channels Bit 2 - SC-QAM Video channels Bit 3 - NDF channels Bit 4 - 55-1 channels Bit 5 - 55-2 channels

B.5.3.7.9 BufferDepthConfigurationSupport[71]

The RPD communicates its capability to support for configuration of the output buffer depth through this TLV. This capability is only applicable to DOCSIS downstream channels.

TLV Type	Length	Units	Access	Value
50.30	1	N/A	R	An unsigned byte, a bitmask specifying the capability of the RPD to support configuration of Output Buffer Depth. If set to '1' the bits listed below indicate the capability to configure output buffer depth on the corresponding types of downstream channels. Bit 0 - OFDM channels Bit 1 - SC-QAM DOCSIS channels

B.5.3.7.10 RpdUcdProcessingTime[72]

This TLV reports the minimum interval needed by the RPD to process a UCD message received via GCP. This interval is equivalent to CM UCD processing time defined in [MULPIv3.1] but its duration can be longer.

TLV Type	Length	Units	Access	Value
50.31	2	microseconds	R	An unsigned short value specifying the minimum interval that the RPD requires to process a UCD message. The maximum value of the RPD UCD Processing time is 50000 usec. The minimum value RPD UCD Processing time is equal to CM UCD processing time (1500 usec for each changed SC-QAM channel or 2000 usec for each changed upstream OFDMA channel) defined in [MULPIv3.1].

[70] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[71] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[72] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

B.5.3.7.11 RpdUcdChangeNullGrantTime[73]

This TLV reports the minimum Null grant interval needed by the RPD in the first MAP with incremented UCD change count. The RPD uses the Null grant in the first map to programs registers of its burst receiver during this interval.

TLV Type	Length	Units	Access	Value
50.32	2	microseconds	R	An unsigned short value specifying the minimum Null grant interval that the RPD requires in the first MAP with incremented UCD change count. The maximum value of the RPD UCD Change Null Grant Time is 4000 usec for each changed channel. The minimum value of the RPD UCD Change Null Grant Time is defined in [MULPIv3.1].

B.5.3.7.12 SupportMultiSectionTimingMerReporting[74]

This object allows the RPD to indicate whether it supports Multi-Section Timing and MER reporting as opposed to just reporting a single average Timing and MER. More detail is documented in the UEPI Probe Pseudowire format of [R-UEPI]. If the RPD is capable, then the configuration of the Multi-Section to subcarrier mapping is made through the "ConfigMultiSectionTimingMer" TLV.

TLV Type	Length	Units	Access	Value
50.33	1	N/A	R	An unsigned byte value indicating whether the RPD supports Multi-section Timing and MER reporting and the flexibility of that reporting 0 - The RPD does not support Multi-Section Timing and MER Reporting 1 - The RPD supports equally spaced non-overlapping sections. 2 - The RPD supports fully flexible sections and spacing of non-overlapping sections.

B.5.3.8 *RPD RDTI Capabilities*

B.5.3.8.1 RdtiCapabilies[75]

RdtiCapabilies TLV allows the RPD to inform the CCAP Core about it its capabilities related to time synchronization.

TLV Type	Length	Units	Access	Value
50.34	variable		N/A	A complex TLV grouping sub-TLVs used to convey RPD's timing capabilities.

B.5.3.8.2 NumPtpPortsPerEnetPort

This TLV allows the RPD to inform the CCAP Core how many PTP ports it support per CIN facing Ethernet Port. If the RPD does not support PTP port configuration from the CCAP Core, as it can be in the case of PHY Shelf device, the RPD reports zero.

TLV Type	Length	Units	Access	Value
50.30.1	1		R	The number of PTP ports supported by the RPD per CIN facing Ethernet Port.

[73] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[74] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[75] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

B.5.3.8.3 MaxDsPspSegCount[76]

This attribute allows the RPD to indicate how many PSP segments it can support in a packet received on downstream PSP pseudowires. Vendor's implementations may restrict the number of PSP segments that can be processed in a packet received on a downstream pseudowire.

The RPD MUST support at minimum 10 PSP segments per packet received on a downstream PSP pseudowire.

The CCAP Core MUST limit the number of PSP segments in packets transmitted on any downstream PSP pseudowire to not exceed the value advertised by the RPD in MaxDsPspSegCount capability.

TLV Type	Length	Units	Access	Value
50.35	1	N/A	R	An unsigned byte value indicating how many PSP segments can the RPD support in packet received on downstream PSP pseudowires. The valid value range is 10-255.

B.5.3.8.4 DirectDsFlowQueueMapping[77]

This attribute allows the RPD to indicate whether it supports direct mapping of downstream PSP flows to strict priority queues. Additional information about direct mapping of downstream PSP flows to strict priority queues can be found in [R-DEPI].

TLV Type	Length	Units	Access	Value
50.36	1	N/A	R	An unsigned byte value indicating whether the RPD supports direct mapping of downstream PSP flows to strict priority queues. 0 - The RPD does not support direct mapping of downstream PSP flows to strict priority queues. 1 - The RPD supports direct mapping of downstream PSP flows to strict priority queues.

B.5.3.8.5 DsSchedulerPhbIdList[78]

The RPD communicates the list of PHB-IDs supported by its downstream scheduler via this attribute. Additional information about PHB-IDs for RPD's downstream scheduler can be found in [R-DEPI].

TLV Type	Length	Units	Access	Value
50.37	variable	N/A	R	A hexadecimal string in which six LSBs of each byte contain a single PHB-ID that is supported by the RPD's scheduler.

B.5.3.8.6 RpdPendingEvRepQueueSize[79]

This attribute permits the RPD to report the size of its Pending Event Report Queue.

TLV Type	Length	Units	Access	Value
50.38	2	Number of Event Reports.	R	An unsigned short value specifying the size of RPD's Pending Event Report Queue.

B.5.3.8.7 RpdLocalEventLogSize[80]

This attribute allows the RPD to report the size of its Local Event Log. The RPD MUST support at minimum Local Event Log size of 20 entries.

[76] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[77] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[78] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[79] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[80] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
50.39	2	Number of Event Reports.	R	An unsigned short value specifying the size of RPD's Local Event Log.

B.5.4 RPD Operational Configuration[81]

The object model representing RPD operational configuration is shown in Figure 31.

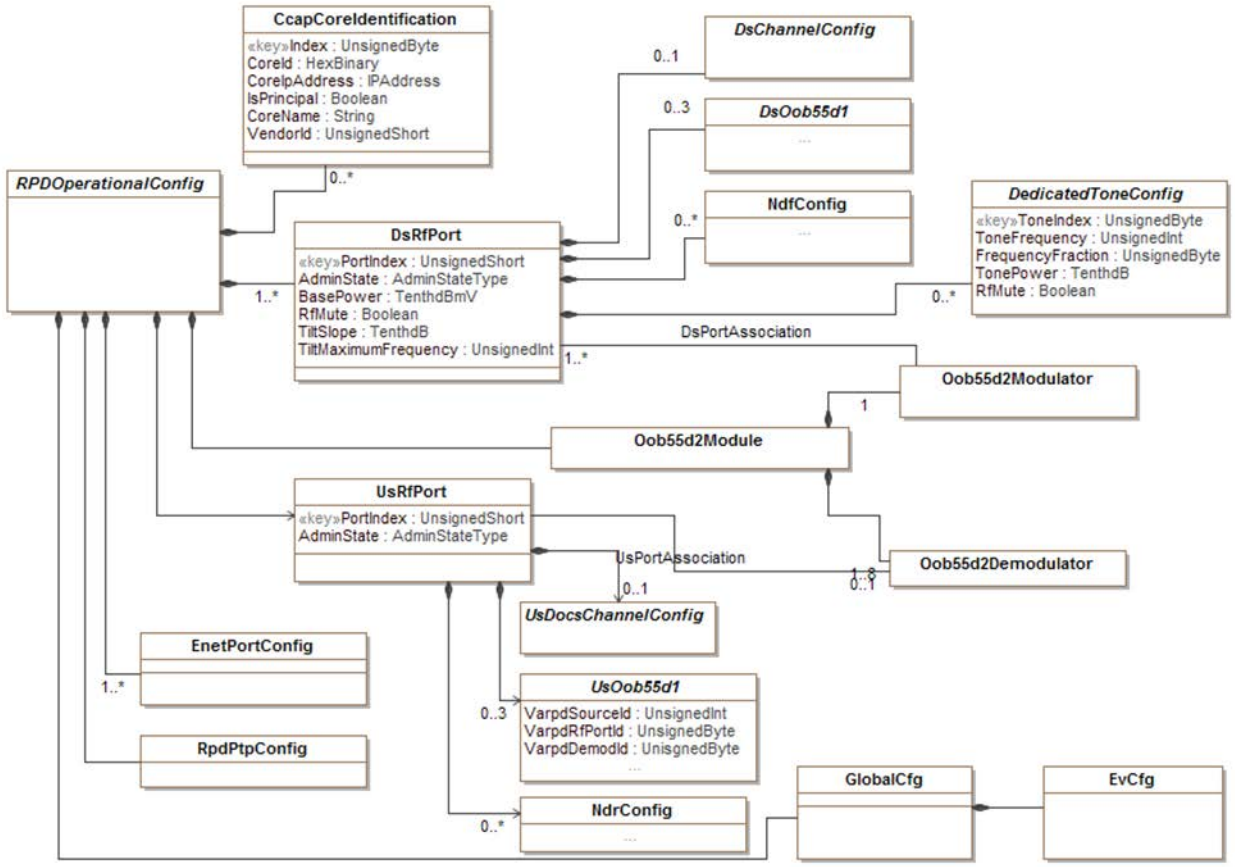

Figure 31 - RPD Operational Configuration Objects [82]

B.5.4.1 *EvCfg*

EvCfg is a complex TLV which is used by the Principal CCAP Core to configure event reporting in the RPD. EvCfg is a sub-TLV of GlobalCfg TLV.

TLV Type	Length	Units	Access	Value
15.1	Variable		N/A	A set of sub-TLV with attributes defining RPD's event reporting configuration. EvCfg can include a number of EvControl sub-TLVs; one for each configured or reported priority level.

[81] Added per R-PHY-N-15.1360-4 on 9/22/15 by JB. Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.
[82] Revised per R-PHY-N-16.1451-1 on 4/15/16 and per R-PHY-N-16.1477-1 on 4/19/16 by JB. Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

B.5.4.1.1 EvControl

EvControl is a complex TLV which is used by the Principal CCAP Core to enable event reporting for the specified event priority in the RPD.

TLV Type	Length	Units	Access	Value
15.1.1	Variable		N/A	A set of two sub-TLV with attributes enabling RPD's event reporting for one event priority. The set of sub-TLV can includes exactly one pair of EvPriority and EvReporting TLVs.

B.5.4.1.2 EvPriority

This attribute is used as an index to select event priority level when the Principal CCAP Core configures event reporting in the RPD. The EvPriority TLV is the first followed immediately by EvReporting TLV. When writing, the CCAP Core MUST send EvPriority and EvReporting attributes as a pair.

TLV Type	Length	Units	Access	Value
15.1.1.1	1		N/A	An unsigned byte value specifying the priority level for event reports for which the configuration is applicable. The valid values are defined in [RFC 4639] and listed below for easier reference: 1 - emergency 2 - alert 3 - critical 4 - error 5 - warning 6 - notice 7 - information 8 - debug

B.5.4.1.3 EvReporting

This attribute configures the handling of the RPD event reports for a selected priority level. EvReporting TLV paired with EvPriority TLV, permits the Principal CCAP Core to selectively control event reports. The RPD MUST preserve the setting of this attribute across reboots in non-volatile memory.

TLV Type	Length	Units	Access	Value
15.1.1.2	1		R/W	An unsigned byte value, a bitmask configuring event reporting by the RPD for the selected priority level. Bit 0 - If set to 0, events reports for selected event priority are not stored in the RPD's Local Event Log. If set to 1, events reports for selected event priority are stored in RPD's Local Event Log. The default value is 1 for priorities 1-4 and zero for other priorities. Bit 1 - If set to 0 event reports for selected priority are not sent to the CCAP Core via GCP Notification. If set to 1 events of selected priority are sent to the CCAP Core via GCP Notification. The default value is 0 for all priorities. All other bits are reserved and set to 0.

B.5.4.1.4 EvThrottleAdminStatus

This attribute controls the transmission of event reports with respect to the event reporting pacing threshold. The four permitted values for are defined in [RFC 4639]. They are applicable only to event reports sent to the CCAP Core via GCP Notify message. Their functions are explained below:

unconstrained (1) causes event reports to be transmitted without regard to the threshold settings.

maintainBelowThreshold (2) causes event reports to be suppressed if the number of event reports would otherwise exceed the threshold configured in EvThrottleThreshold.

stopAtThreshold (3) causes event reports transmission to cease at the threshold and not to resume until EvThrottleAdminStatus is written to again. This setting is primarily used for debugging purposes and need to be used with care as it may result in unreported events.

inhibited (4) causes all event reports to be suppressed.

Writing to this attribute resets the thresholding state. The RPD MUST preserve the setting of this attribute across reboots in non-volatile memory.

TLV Type	Length	Units	Access	Value
15.1.2	1		R/W	An unsigned byte value, The valid values are defined in [RFC 4639] and listed below for easier reference: 1- unconstrained 2 - maintainBelowThreshold 3 - stopAtThreshold 4 - inhibited The default value is 1 (unconstrained).

B.5.4.1.5 EvThrottleThreshold

This attribute specifies permitted number of event reports sent to the CCAP Core per EvThrottleInterval before the RPD applies throttling. The RPD MUST preserve the setting of this attribute across reboots in non-volatile memory.

TLV Type	Length	Units	Access	Value
15.1.3	4		R/W	An unsigned integer which defines permitted number of event reports sent to the CCAP Core per EvThrottleInterval before the RPD applies throttling.

B.5.4.1.6 EvThrottleInterval

This attribute specifies the interval over which EvThrottleThreshold applies. The RPD MUST preserve the setting of this attribute across reboots in non-volatile memory.

TLV Type	Length	Units	Access	Value
15.1.4	4	Seconds	R/W	An unsigned integer value in range 1-2147483647.

B.5.4.1.7 NotifyEnable

This attribute is used to enable the transport of event reports to the Principal CCAP Core via Notify message. The RPD MUST reset value of this attribute to 0 when the RPD initializes, re-initializes or loses the connection to the CCAP Core. The CCAP Core can re-enable reporting of events via Notify message when it is ready to receive them. Such behavior, which requires explicit re-enablement of the transport of event reports via Notify message is intended to avoid a situation when the newly connected RPDs would flood the CCAP Core with accumulated event reports.

TLV Type	Length	Units	Access	Value
15.1.5	1		R/W	An unsigned byte controlling the event report transport over GCP/RCP Notify. The valid values are: 0 - The RPD is not enabled to send event reports via Notify message. 1- The RPD is enabled to send event reports via Notify message. The default value is 0. Note the RPD behavior for this attribute explained above.

B.5.4.2 *CCAP Core Identification*

B.5.4.2.1 Index

This TLV specifies an index to the CCAP Core Identifications table.

TLV Type	Length	Units	Access	Value
60.1	1		N/A	A unsigned byte with a zero based index identifying the CCAP Core associated with the RPD.

B.5.4.2.2 CoreId

This TLV uniquely defines a CCAP Core.

TLV Type	Length	Units	Access	Value
60.2	variable		R/W	A hex-binary string providing unique identification of the CCAP Core, for example a MAC address.

B.5.4.2.3 CoreIpAddress

The IP address of the CCAP Core.

TLV Type	Length	Units	Access	Value
60.3	4 or 16		R/W	The IP address of the CCAP Core. The length signifies whether it's an IPv4 or IPv6 address.

B.5.4.2.4 IsPrincipal

This TLV identifies the CCAP Core as principal.

TLV Type	Length	Units	Access	Value
60.4	1		R/W	A Boolean value identifying the CCAP Core as principal. The valid values are: 0 - CCAP Core is not principal. 1- CCAP Core is Principal.

B.5.4.2.5 CoreName

This TLV identifies the CCAP Core by name.

TLV Type	Length	Units	Access	Value
60.5	variable		R/W	A string containing the CCAP Core's name.

B.5.4.2.6 VendorId

This TLV identifies the manufactured of the CCAP Core.

TLV Type	Length	Units	Access	Value
60.6	2		R/W	An unsigned short number containing the vendor identification of the CCAP Core.

B.5.4.3 *Downstream RF Port Configuration*

B.5.4.3.1 DsRfPort

DsRfPort This complex TLV specifies is used to communicate configuration information related to downstream RF port.

TLV Type	Length	Units	Access	Value
61	variable		N/A	One or more sub-TLVs.

B.5.4.3.2 AdminState

This TLV describes the administrative state for the RF Port.

TLV Type	Length	Units	Access	Value
61.2	1	N/A	R/W	The administrative state of the RF Port. The defined values are: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.4.3.3 BasePower

This TLV allows the CCAP Core to configure the base output power for each downstream channel on the RPD DS RF port. The value is expressed in units of TenthdBmV. Acceptable power ranges for this attribute are defined in [DRFI] in section 6.3.5.1.1, Power per Channel CMTS or EQAM.

TLV Type	Length	Units	Access	Value
61.3	1	TenthdBmV	R/W	The base output power for each downstream channel on the RPD DS RF port.

B.5.4.3.3.1 RfMute

This TLV allows the CCAP Core to mute an RF port on the RPD. This is a diagnostic state. It only affects the output power of the RF port. The operational status of any channel on the port is not affected.

TLV Type	Length	Units	Access	Value
61.4	1		R/W	A Boolean value indicating whether the RF port is muted. 0 - Port is not muted. 1 - Port is muted. Values 2-255 are reserved.

B.5.4.3.3.2 TiltSlope

This TLV configures the tilt slope value to be applied to the downstream spectrum on the RPD.

TLV Type	Length	Units	Access	Value
61.5	1	TenthdB	R/W	Tilt slope value to be applied to the DS spectrum of the selected port of the RPD.

B.5.4.3.4 TiltMaximumFrequency[83]

This TLV configures the frequency of the tilt pivot point. Tilt pivot point is the maximum frequency point where the Tilt Slope is applicable.

TLV Type	Length	Units	Access	Value
61.6	4	Hertz	R/W	The frequency of the tilt pivot point.

B.5.4.3.5 DedicatedToneConfig

This complex TLV is used to configure the dedicated CW tone generators in the RPD. The RPD retains the CW tone configuration in its non-volatile memory and restores it immediately after reboot/restart.

[83] Revised Sections B.6.4.2.5 through B.6.4.2.9 per R-PHY-N-16.1451-1 on 4/15/16 by JB.

TLV Type	Length	Units	Access	Value
61.7	variable	N/A		A set of TLV with attributes of dedicated tone generator.

B.5.4.3.6 ToneIndex

This TLV select the index of the tone generator in the RPD. The sender MUST include the Tone Index TLV as the first sub-TLV of the DedicatedToneConfig TLV.

TLV Type	Length	Units	Access	Value
61.7.1	1	N/A	N/A	The selector of the dedicator pilot tone generator. The valid range is from 0 to NumCwToneGens - 1.

B.5.4.3.7 ToneFrequency

This TLV is used to configure the frequency of CW tone in the dedicated pilot tone generator in the RPD.

TLV Type	Length	Units	Access	Value
61.7.2	4	Hertz	R/W	An unsigned integer specifying the frequency of the CW tone.

B.5.4.3.8 TonePower

This TLV is used to configure the power level of CW carrier output from the dedicated pilot tone generator in the RPD

TLV Type	Length	Units	Access	Value
61.7.3	1	TenthDb	R/W	A signed byte value specifying the power of the CW carrier relative to QAM-256 channel power.

B.5.4.3.9 RfMute

RfMute TLV is used to mute the output of the CW tone generator. If set to 1, the generator is in the muted diagnostic state i.e., transmitting no signal.

TLV Type	Length	Units	Access	Value
61.7.4	1	N/A	R/W	A Boolean value which specifies whether the selected generator is in the muted state. 0 - Generator is not muted. 1 - Generator is muted. Values 2-255 are reserved

B.5.4.3.10 FrequencyFraction

FrequencyFraction TLV is used to configure the fractional frequency of CW tone of the dedicated pilot tone generator in the RPD. This TLV allows the CCAP Core to specify the frequency of the CW tone with additional precision at the level of 0.1 Hertz.

TLV Type	Length	Units	Access	Value
61.7.5	1	TenthHz	R/W	An unsigned byte value in range 0-9 specifying the fractional frequency of the CW tone in units of tenths of Hertz. Default value is 0.

B.5.4.4 *DOCSIS and MPEG Video Downstream Channel Configuration*

The RPD's configuration object model for downstream DOCSIS and MPEG video channels is presented in Figure 32. The diagram includes the configuration objects of DOCSIS SC-QAM channels/MPEG Video channels (grouped in DsScQamChannelConfig object) and OFDM channels (grouped in DsQamChannelConfig object).

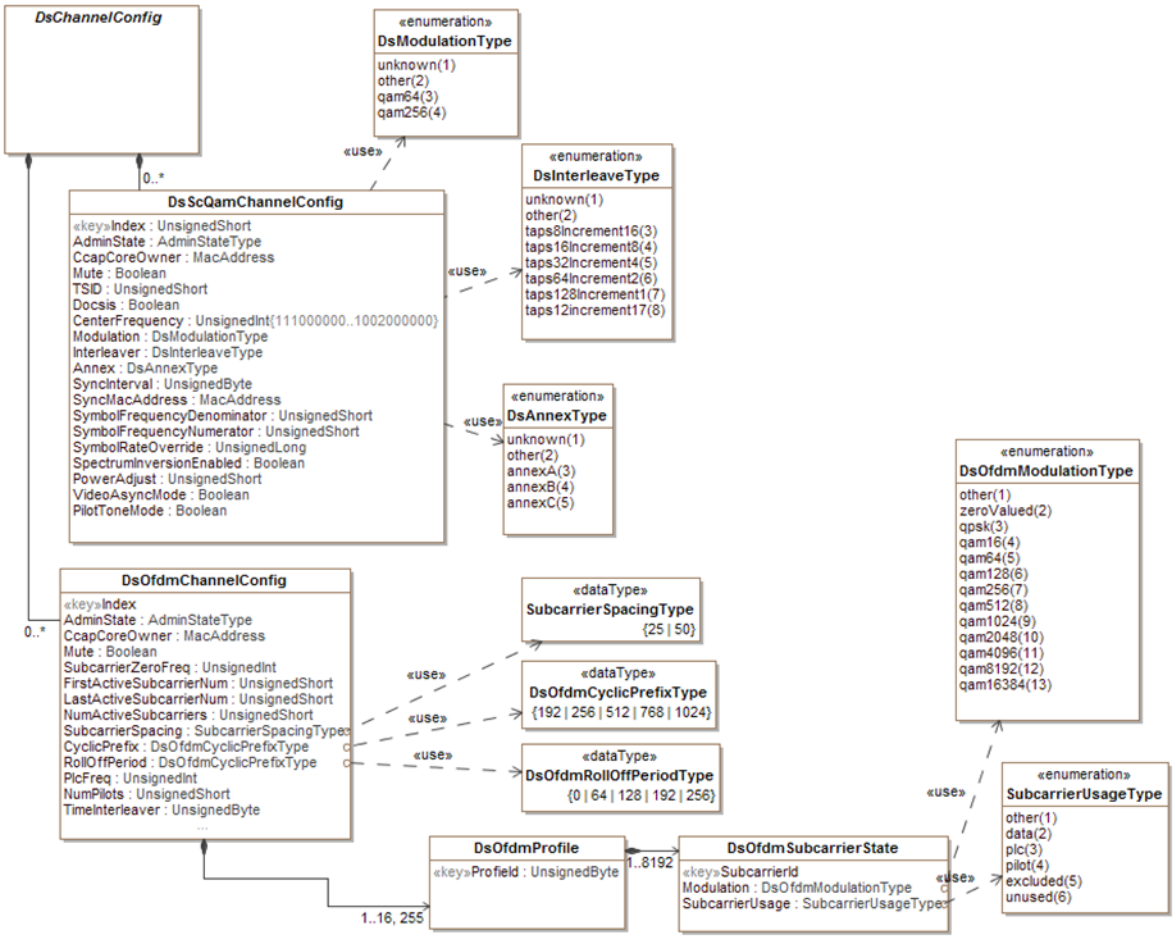

Figure 32 - RPD DOCSIS and MPEG Video Downstream Channel Configuration

B.5.4.5 *Downstream SC-QAM Channel Configuration TLVs.*

B.5.4.5.1 AdminState

This object describes the administrative state of the QAM channel.

TLV Type	Length	Units	Access	Value
62.1	1		R/W	The administrative state of the QAM Channel. The AdminState can have possible values: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.4.5.2 CcapCoreOwner

This TLV specifies the MAC address of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
62.2	6		R/W	The MAC address of the CCAP Core operating the channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.4.5.3 RfMute

This object configures the mute state of the downstream QAM channel. If set to true, the channel is in the muted state.

TLV Type	Length	Units	Access	Value
62.3	1		R/W	A Boolean value which specifies whether the selected channel is in the muted state. 0 - Channel is not muted. 1 - Channel is muted. Values 2-255 are reserved.

B.5.4.5.4 TSID

This TLV specifies the Transport Stream Identifier (TSID) for the MPEG Video QAM channel. This TLV is not used for DOCSIS channels.

TLV Type	Length	Units	Access	Value
62.4	2		R/W	Transport stream ID for the QAM channel. This value can be optionally configured by the CCAP Core to help with debug.

B.5.4.5.5 CenterFrequency

This TLV specifies the center frequency of the channel, in Hz or the frequency of the CW tone, when a channel is configured as Pilot Tone or Alignment Carrier.

TLV Type	Length	Units	Access	Value
62.5	4	Hertz	R/W	An unsigned long specifying the center frequency of a QAM channel or the frequency of the CW tone.

B.5.4.5.6 OperationalMode[84]

This TLV specifies the mode in which a QAM channel resource is operating.

TLV Type	Length	Units	Access	Value
62.6	1		R/W	An enumerated value specifying the operation mode in which the channel is operating. Defined values are: 1 - Other 2 - Channel operates as DOCSIS channel. 3 - Channel operates as a synchronous MPEG video channel. 4 - Channel operates as an asynchronous MPEG video channel. 5 - Channel operates as CW carrier; that is as a Pilot Tone or an Alignment Carrier. Values 0, 5-255 are reserved.

[84] Revised per R-PHY-N-16.1451-1 on 4/15/16 by JB

B.5.4.5.7 Modulation

This TLV specifies the QAM modulation order for the QAM channel.

TLV Type	Length	Units	Access	Value
62.7	1		R/W	An enumerated value defining QAM modulation order. The defined values are: 1 - Unknown 2 - Other 3 - Qam64 4 -Qam256 Values 5-255 are reserved.

B.5.4.5.8 InterleaverDepth

This TLV specifies interleaver depth of a channel.

TLV Type	Length	Units	Access	Value
62.8	1		R/W	An enumerated value defining the depth of the interleaver. It has the following defined values: 1 - unknown(1) 2 - other(2) 3 - taps8Increment16(3) 4 - taps16Increment8(4) 5 - taps32Increment4(5) 6 - taps64Increment2(6) 7 - taps128Increment1(7) 8 - taps12increment17(8) Values 9-255 are reserved.

B.5.4.5.9 Annex

This TLV specifies the Annex type of the downstream channel.

TLV Type	Length	Units	Access	Value
62.9	1		R/W	Annex type of the channel. The defined values are: 1 - Unknown 2 - Other 3 - annex A 4 - annex B 5 - annex C Values 0, 6-255 are not reserved.

B.5.4.5.10 SyncInterval

This TLV specifies the interval between the SYNC messages sent on DOCSIS SC-QAM channel.

TLV Type	Length	Units	Access	Value
62.10	1	Milliseconds	R/W	The interval between the SYNC messages. Valid operational values are 5 msec to 200 msec. Value of 0 signifies that SYNC messages are not sent.

B.5.4.5.11 SyncMacAddress

This TLV specifies the source MAC address for the SYNC messages sent on DS channel.

TLV Type	Length	Units	Access	Value
62.11	6		R/W	The MAC address which the RPD need to insert into the SYNC messages.

The RPD MUST NOT send SYNC messages until the CCAP configures the MAC address.

B.5.4.5.12 SymbolFrequencyDenominator

This TLV specifies the denominator (N) of the M/N ratio relating the downstream symbol clock to the DOCSIS 10.24MHz clock.

TLV Type	Length	Units	Access	Value
62.12	2		R/W	An unsigned integer value which is the denominator (N) of the M/N ratio relating the downstream symbol clock to the DOCSIS 10.24MHz clock.

B.5.4.5.13 SymbolFrequencyNumerator

This TLV specifies the numerator (M) of the M/N ratio relating the downstream symbol clock to the DOCSIS 10.24MHz clock.

TLV Type	Length	Units	Access	Value
62.13	2		R/W	An unsigned integer value which is the numerator (M) of the M/N ratio relating the downstream symbol clock to the DOCSIS 10.24MHz clock.

B.5.4.5.13.1 SymbolRateOverride

This TLV specifies a symbol rate to be used in selected cases MPEG video QAM channels. This parameter is not applicable to DOCSIS downstream QAM channels.

TLV Type	Length	Units	Access	Value
62.14	4	Unsigned Long	R/W	An unsigned integer which specifies a downstream symbol rate to be used by the RPD on the channel.

B.5.4.5.14 SpectrumInversionEnabled

This TLV specifies RF signal spectrum inversion. When set to true, the MPEG video QAM channel's spectrum is inverted. This parameter is not applicable to DOCSIS downstream SC-QAM channels.

TLV Type	Length	Units	Access	Value
62.15	1		R/W	A Boolean value which specifies whether the channel spectrum is inverted. 0 - Channel's spectrum is not inverted. 1 - Channel's spectrum is inverted. Values 2-255 are reserved.

B.5.4.5.15 PowerAdjust

This object specifies power adjustment for the channel.

TLV Type	Length	Units	Access	Value
62.16	1	TenthdB	R/W	A signed value power adjustment amount in units of 0.1 dB relative to the base power level specified for the DS RF port.

B.5.4.6 *Configuration Objects for a Downstream OFDM Channel*[85]

The RCP configuration objects for a downstream OFDM channel are grouped in DsOfdmChannelConfig object.

[85] Revised per R-PHY-N-16.1573-3 on 9/8/16 by JB.

The CCAP Core configures the majority of the OFDM channel parameters by sending the OCD message to the RPD. Those parameters which can be included in the OCD message are defined below as real-only.

TLV Type	Length	Units	Access	Value
63	variable			The set of sub-TLVs representing the configuration parameters of the OFDM channel.

B.5.4.6.1 AdminState

This TLV describes the administrative state for the selected OFDM channel.

TLV Type	Length	Units	Access	Value
63.1	1		R/W	The administrative state of the OFDM Channel. The following values of the AdminState have been defined: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.4.6.2 CcapCoreOwner

This TLV specifies the MAC address of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
63.2	6		R/W	The MAC address of the CCAP Core operating the channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.4.6.3 RfMute

This TLV configures the mute state of the downstream OFDM channel. If set to true, the channel is in the muted state.

TLV Type	Length	Units	Access	Value
63.3	1		R/W	A Boolean value which specifies whether the selected channel is in the muted state. 0 - Channel is not muted. 1 - Channel is muted. Values 2-255 are reserved.

B.5.4.6.4 SubcarrierZeroFreq

This TLV specifies the frequency of subcarrier zero of the OFDM channel.

TLV Type	Length	Units	Access	Value
63.4	4	Hertz	R	The frequency of subcarrier zero.

B.5.4.6.5 FirstActiveSubcarrier

This TLV specifies the first active subcarrier of the OFDM channel.

TLV Type	Length	Units	Access	Value
63.5	2		R	The unsigned number identifying first active subcarrier.

B.5.4.6.6 LastActiveSubcarrier

This TLV specifies the highest numbered active subcarrier of the OFDM channel.

TLV Type	Length	Units	Access	Value
63.6	2		R	The unsigned number identifying the last active subcarrier.

B.5.4.6.7 NumActiveSubcarriers

This TLV specifies the number of active subcarriers of the OFDM channel.

TLV Type	Length	Units	Access	Value
63.7	2		R	An unsigned number specifying the total number active subcarriers of the OFDM channel.

B.5.4.6.8 CyclicPrefix

This TLV specifies represents the cyclic prefix setting, which enables the OFDM 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 CP and the choice depends on the delay spread of the channel - a longer delay spread requires a longer cyclic prefix.

TLV Type	Length	Units	Access	Value
63.8	1		R	Valid values and their corresponding sample values are: 1 - 192 2 - 256 3 - 512 4 - 768 5 - 1024 Other values are reserved.

B.5.4.6.9 RollOffPeriod

This TLV specifies represents the roll off period or windowing which maximizes channel capacity by sharpening the edges of the spectrum of the OFDM signal.

TLV Type	Length	Units	Access	Value
63.9	1		R	The type of the OFDM windowing 1 - 0 2 - 64 3 - 128 4 - 192 5 - 256 Other values are reserved.

B.5.4.6.10 PLCFreq

This TLV specifies the center frequency of the lowest subcarrier of the 6 MHz encompassed spectrum containing the PLC at its center. The frequency of this subcarrier is required to be located on a 1 MHz grid.

Type	Length	Units	Access	Value
63.10	4	Hertz	R	An unsigned number specifying the center frequency of the lowest subcarrier of the 6 MHz encompassed spectrum containing the PLC at its center.

B.5.4.6.11 TimeInterleaverDepth

This TLV specifies the depth of time interleaver of the OFDM channel.

TLV Type	Length	Units	Access	Value
63.11	1		R	A number in range of 1-32 specifying the depth of the time interleaver in symbols.

B.5.4.6.12 SubcarrierSpacing

This TLV specifies the subcarrier spacing of the OFDM channel

TLV Type	Length	Units	Access	Value
63.12	1		R	A value indicating subcarrier spacing. Valid values are: 1 - Subcarrier spacing of 25 KHz. 2 - Subcarrier spacing of 50 KHz. All other values are reserved.

B.5.4.6.13 DsOfdmSubcarrierType

TLV Type	Length	Units	Access	Value
63.13	Variable		N/A	A set of sub-TLVs defining the subcarrier usage for ranges of subcarriers.

B.5.4.6.13.1 StartSubcarrierId

This TLV specifies the first subcarrier if in a range of subcarrier ids.

TLV Type	Length	Units	Access	Value
63.13.1	2		N/A	A number within the range 0-8191.

B.5.4.6.13.2 EndSubcarrierId

This TLV specifies the last subcarrier id in a range of subcarrier ids.

TLV Type	Length	Units	Access	Value
64.13.2	2		N/A	A number within the range 0-8191.

B.5.4.6.13.3 SubcarrierUsage

This TLV specifies the configured usage for a range of subcarriers.

TLV Type	Length	Units	Access	Value
				The value of the TLV specifies the type of subcarrier.
63.13.3	1		R	The defined values are: 1 - Other 2 - Data 3 - Plc 4 - Continuous Pilot 5 - Excluded 6 - Unused All other values are reserved.

B.5.4.7 *DsOfdmProfile TLVs*[86]

The modulation levels of the subcarriers in a channel are grouped into DsOfdmProfile TLV. The CCAP Core configures all OFDM profile parameters by sending the DPD message to the RPD. The parameters which are included in DsOfdmProfile TLV are defined below as real-only.

TLV Type	Length	Units	Access	Value
64	Variable		N/A	A set of sub-TLVs with parameters describing a single OFDM profile. The TLV includes exactly one ProfileId sub-TLV and one or more DsOfdmSubcarrierModulation sub-TLVs.

B.5.4.7.1 ProfileId[87]

The ProfileID TLV selects a profile ID of the downstream OFDM channel. The ProfileID TLV appears exactly one time in the DsOfdmProfile TLV. The sender of the RCP message MUST include the ProfileID TLV as the first sub-TLV of the DsOfdmProfile TLV.

TLV Type	Length	Units	Access	Value
64.1	1		N/A	An unsigned number within the range 0-15 or 255. OFDM Profile ID of the selected OFDM channel.

B.5.4.7.2 DsOfdmSubcarrierModulation

This TLV specifies modulation level for a range of data subcarriers for a particular profile. This TLV can appear multiple times in the DsOfdmProfile TLV.

TLV Type	Length	Units	Access	Value
64.2	Variable		N/A	A set of TLVs specifying the modulation order for a range of subcarriers.

B.5.4.7.2.1 StartSubcarrierId

This TLV specifies the first subcarrier id in a subcarrier range or subcarrier ids.

TLV Type	Length	Units	Access	Value
64.2.1	2		N/A	A number within the range 0-8191.

B.5.4.7.2.2 EndSubcarrierId

This TLV specifies the last subcarrier id in a range of subcarrier ids.

TLV Type	Length	Units	Access	Value
64.2.2	2		R	A number within the range 0-8191.

B.5.4.7.2.3 Modulation

This TLV describes the modulation level assigned to a range of data subcarriers.

[86] Revised per R-PHY-N-16.1573-3 on 9/8/16 by JB.
[87] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
64.2.3	1	DsOfdmModulationType	R	The defined values are: 1- Other 2 - zeroValued 3 - qpsk 4 - qam16 5 - qam64 6 - qam128 7- qam256 8 - qam512 9 - qam1024 10 - qam2048 11- qam4096 12 - qam8192 13- qam16384 All other values are reserved.

B.5.4.8 *DOCSIS Upstream Channel Configuration*

The RPD's configuration object model for upstream DOCSIS channels is presented in Figure 33, the diagram group configuration of DOCSIS SC-QAM channels and OFDMA channels. The CCAP Core configures the majority of the upstream channel parameters by sending the UCD message to the RPD. Those upstream channel parameters which can be included in the UCD message are defined below as real-only.

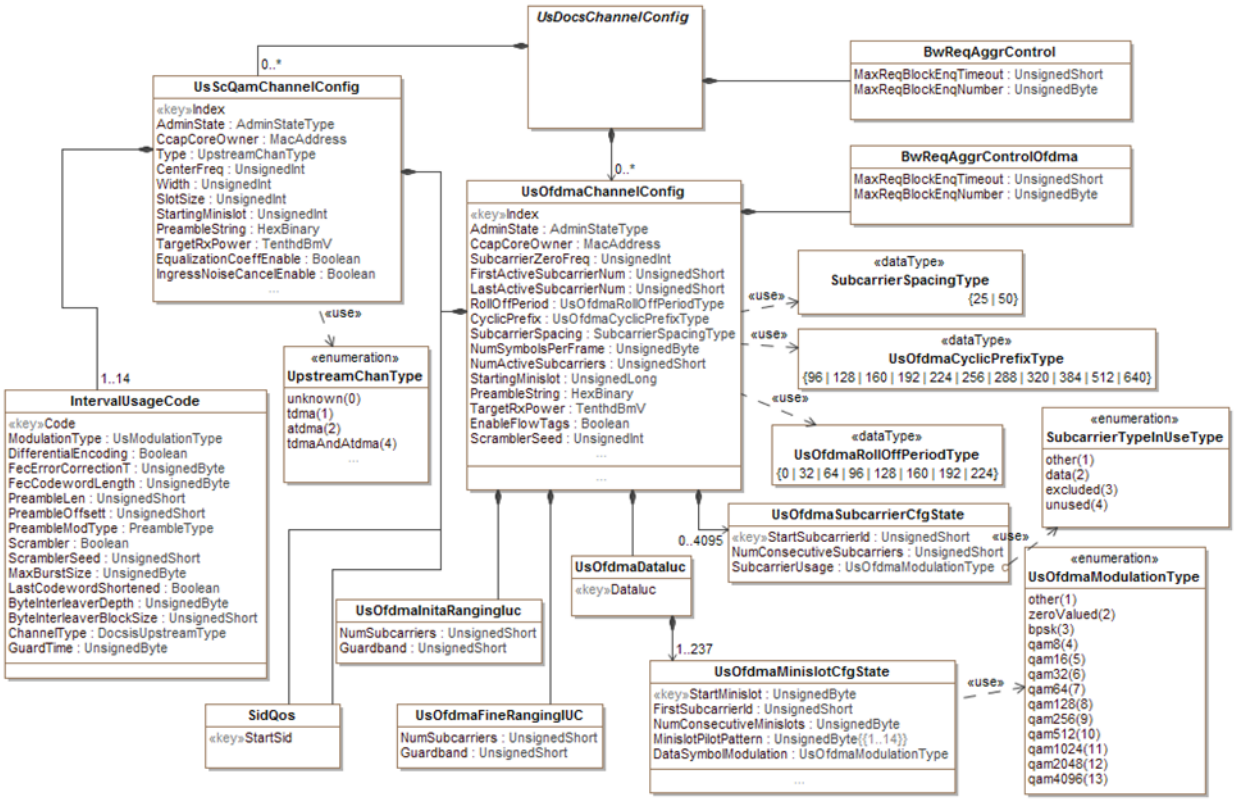

***Figure 33 - DOCSIS Upstream Channel Configuration*[88]**

[88] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

B.5.4.9 *UsScQamChannelConfig TLVs*

B.5.4.9.1 AdminState

This TLV describes the administrative state of the SC-QAM channel.

TLV Type	Length	Units	Access	Value
65.1	1		R/W	The administrative state of the SC-QAM channel. The possible values are: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.4.9.2 CcapCoreOwner

This TLV specifies the MAC address of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
65.2	6	MacAddress	R/W	The MAC address of the CCAP Core operating the channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.4.9.3 TLV Type

This TLV specifies the upstream QAM channel type.

TLV Type	Length	Units	Access	Value
65.3	1		R	The defined values are: 0 - Unknown(0) 1 - TDMA 2 - ATDMA 3 - Reserved 4 - TDMAandATDMA All other values are reserved.

B.5.4.9.4 CenterFrequency

This TLV specifies the center frequency of the channel in Hz.

TLV Type	Length	Units	Access	Value
65.4	4	Hertz	R	Center frequency of the channel.

B.5.4.9.5 Width

This TLV specifies the width of the upstream SC-QAM channel in Hz.

TLV Type	Length	Units	Access	Value
65.5	4	Hertz	R	The width of the upstream QAM channel in Hz. The permitted values are: 200,000, 400,000, 800,000, 1,600,000, 3,200,000, 6,400,000.

B.5.4.9.6 SlotSize

This TLV specifies the channel's minislot size as a number of 6.25 usec ticks.

TLV Type	Length	Units	Access	Value
65.6	4	6.25 usec tics	R	Minislot size expressed as a number of 6.25 usec tics.

B.5.4.9.7 StartingMinislot[89]

When written by the CCAP Core this TLV specifies the future time (expressed as a 32-bit DOCSIS timestamp value) when the upstream channel change (signaled by MAP messages to CMs) goes into effect. The 32-bit timestamp points to the Alloc Start Time of the first MAP with incremented UCD Count. When read, the RPD return zero.

TLV Type	Length	Units	Access	Value
65.7	4		R/W	An unsigned 32-bit value specifying the future time expressed as the a DOCSIS timestamp value when the most recent channel configuration change will go into effect.

B.5.4.9.8 PreambleString

This TLV defines the complete preamble pattern superstring including standard and extended preamble pattern string from UCD. The maximum length of the string is 128+64 = 192. The string is formatted in conformance to the MULPI specification.

TLV Type	Length	Units	Access	Value
65.8	variable		R	The complete preamble pattern string including standard and extended preamble pattern string from UCD.

B.5.4.9.9 TargetRxPower

This TLV specifies the desired receive power level, in TenthdBmV for the channel.

TLV Type	Length	Units	Access	Value
65.9	2	TenthdBmV	R/W	The desired receive power level, in TenthdBmV for the channel.

B.5.4.9.10 EqualizationCoeffEnable

This attribute can be used by the CCAP Core to suppress sending equalization coefficients by the RPD with ranging bursts. The primary purpose of this attribute is to aid in debugging of upstream issues.

TLV Type	Length	Units	Access	Value
65.11	1	N/A	R/W	An unsigned byte value which can enable/disable sending pre-equalization coefficients in ranging bursts. The valid values are: 0 - Sending of EQ coefficient is suppressed. 1 - Sending of EQ coefficient is not suppressed. All other values are reserved. Default value is 1.

B.5.4.9.11 IngressNoiseCancelEnable

This attribute can be used by the CCAP Core to enable ingress noise cancellation function for the channel in the RPD.

[89] Revised per R-PHY-N-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
65.12	1	N/A	R/W	An unsigned byte value which can enable/disable enable ingress noise cancellation function for the channel in the RPD. The valid values are: 0 - Ingress noise cancellation is not enabled. 1 - Ingress noise cancellation is enabled. All other values are reserved. Default value is 0.

B.5.4.10 *Interval Usage Code TLVs*

B.5.4.10.1 Code

This TLV identifies the Interval Usage Code (IUC).

TLV Type	Length	Units	Access	Value
65.10.1	1		N/A	A number identifying the IUC. Valid values are 1-14. All other values are reserved.

B.5.4.10.2 DifferentialEncoding

This TLV specifies whether differential encoding is enabled.

TLV Type	Length	Units	Access	Value
65.10.2	1		R	A Boolean value which indicates whether differential encoding is on. The valid values are: 0 - Differential Encoding is off. 1- Differential Encoding is on.

B.5.4.10.3 FecErrorCorrectionT

This TLV specifies the mode of Reed-Solomon FEC mode.

TLV Type	Length	Units	Access	Value
65.10.3	1		R	The mode of R-S FEC mode. The defined values are 0-16. 0 implies no FEC.

B.5.4.10.4 FecCodewordLength

This TLV specifies Reed-Solomon FEC codeword length (k) information bytes.

TLV Type	Length	Units	Access	Value
65.10.4	1	bytes	R	An unsigned number specifying the R-S FEC codeword information bytes. Valid values are 16 to 253 for fixed and shortened codewords.

B.5.4.10.5 PreambleLen

This TLV specifies the length of the preamble for the IUC in bits.

TLV Type	Length	Units	Access	Value
65.10.5	2	bits	R	An unsigned number specifying the length of the preamble for an IUC.

B.5.4.10.6 PreambleOffset

This TLV specifies the starting offset into the preamble superstring, for which the bits to be used in the preamble.

TLV Type	Length	Units	Access	Value
65.10.6	2	bits	R	An unsigned number specifying the starting offset into the preamble superstring, which point to be preamble bits to be used in the IUC.

B.5.4.10.7 PreambleModType

This TLV specifies whether QPSK0 or QPSK1 is used for preamble.

TLV Type	Length	Units	Access	Value
65.10.7	1		R	A value specifying the modulation type for preamble. The defined values are: 1 - QPSK0 2 - QPSK1 All other values are reserved.

B.5.4.10.8 Scrambler

This TLV specifies whether scrambler is used for the IUC.

TLV Type	Length	Units	Access	Value
65.10.8	1		R	A Boolean value specifying whether scramble is enabled for the IUC. Defined values are: 0 - Scrambler is off. 1- Scrambler is on.

B.5.4.10.9 ScramblerSeed

This TLV specifies the scrambler seed. The left most 15 bits are used. The last bit is not used.

TLV Type	Length	Units	Access	Value
65.10.9	2		R	An unsigned value for the scrambler seed. The left most 15 bits are used. The last bit is not used.

B.5.4.10.10 MaxBurstSize

This TLV specifies the maximum number of minislots that can be transmitted during a burst.

TLV Type	Length	Units	Access	Value
65.10.10	1		R	An unsigned number specifying the maximum number of minislots that can be transmitted during a burst.

B.5.4.10.11 LastCodeWordShortened

This TLV specifies whether the last codeword is shortened. 1=fixed (not shortened); 2=shortened.

TLV Type	Length	Units	Access	Value
65.10.11	1		R	A Boolean value specifying whether the last codeword is shortened. Defined values are: 0 - last codeword is fixed (not shortened); 1 - last codeword is shortened. All other values are reserved.

B.5.4.10.12 InterleaverDepth

This TLV specifies the R-S block interleaver depth.

TLV Type	Length	Units	Access	Value
65.10.12	1		R	An unsigned value specifying the R-S block interleaver depth. Defined values are: 0 indicates Dynamic mode; 1 indicates R-S interleaving is disabled.

B.5.4.10.13 ByteInterleaverBlockSize

This TLV specifies the R-s block interleaving size in Dynamic mode.

TLV Type	Length	Units	Access	Value
65.10.13	2		R	The R-S block interleaving size in Dynamic mode.

B.5.4.10.14 ModulationType

This TLV specifies the modulation order for the IUC.

TLV Type	Length	Units	Access	Value
65.10.14	1		R	An unsigned byte value specifying the modulation order for the IUC. 1 - other 2 - QPSK 3 - QAM8 4 - QAM16 5 - QAM32 6 - QAM64 7 - QAM128 All other values are reserved.

B.5.4.10.15 GuardTime

This TLV specifies the number of modulation intervals measured from the end of the last symbol of one burst to the beginning of the first symbol of the preamble of an immediately following burst.

TLV Type	Length	Units	Access	Value
65.10.15	1		R	The number of modulation intervals measured from the end of the last symbol of one burst to the beginning of the first symbol of the preamble of an immediately following burst.

B.5.4.11 *Upstream OFDMA Channel Configuration TLVs*

The upstream OFDMA channel configuration objects are grouped in UsOfdmaChannelConfig object.

B.5.4.11.1 AdminState

This TLV describes the administrative state for the OFDM channel.

TLV Type	Length	Units	Access	Value
66.1	1		R/W	The administrative state of the US OFDMA channel. The possible values are: 1 - “other” 2 - “up” 3 - “down” 4 - “testing” Values 0, 5-255 are reserved.

B.5.4.11.2 CcapCoreOwner

This TLV specifies the MAC address of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
66.2	6	MacAddress	R/W	The MAC address of the CCAP Core operating the channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.4.11.3 SubcarrierZeroFreq

This TLV specifies the frequency of subcarrier zero of the OFDMA channel.

TLV Type	Length	Units	Access	Value
66.3	4	Hertz	R	An unsigned number specifying the frequency of subcarrier zero.

B.5.4.11.4 FirstActiveSubcarrier

This TLV specifies the first active subcarrier of the OFDMA channel.

TLV Type	Length	Units	Access	Value
66.4	2		R	An unsigned number specifying the first active subcarrier.

B.5.4.11.5 LastActiveSubcarrier

This TLV specifies the last active subcarrier of the OFDMA channel.

TLV Type	Length	Units	Access	Value
66.5	2		R	An unsigned number specifying the last active subcarrier.

B.5.4.11.6 RollOffPeriod

This TLV specifies the OFDMA roll-off period for the OFDMA channel with exception for initial ranging bursts.

TLV Type	Length	Units	Access	Value
66.6	2		R	The OFDMA roll-off period. The defined values are: 1 - 0 samples 2 -32 samples 3 - 64 samples 4- 96 samples 5 - 128 samples 6 -160 samples 7 -192 samples 8 -224 samples All other values are reserved.

B.5.4.11.7 CyclicPrefix

This TLV specifies the size of cyclic prefix size.

TLV Type	Length	Units	Access	Value
66.7	2		R	Enumerated value corresponding to cyclic prefix sizes defined in DOCSIS 3.1. The defined values are: 1 - 96 samples 2 - 128 samples 3 - 160 samples 4 - 192 samples 5 - 224 samples 6 - 256 samples 7 - 288 samples 8 - 320 samples 9 - 384 samples 10 - 512 samples 11 -640 samples All other values are reserved.

B.5.4.11.8 SubcarrierSpacing

This TLV specifies the subcarrier spacing for the channel.

TLV Type	Length	Units	Access	Value
66.8	1		R	Enumerated value corresponding to subcarrier spacing defined in DOCSIS 3.1. The valid values are: 1- 25 KHz 2- 50 KHz All other values are reserved.

B.5.4.11.9 NumSymbolsPerFrame[90]

This TLV specifies the number of symbols in a frame.

TLV Type	Length	Units	Access	Value
66.9	1		R	An unsigned number specifying the number of symbols in an OFDMA frame. Valid values are from 6 to 36 with additional restrictions as outlined in the DOCSIS 3.1 PHY specification.

B.5.4.11.10 NumActiveSubcarriers

This TLV describes the number of active subcarriers.

TLV Type	Length	Units	Access	Value
66.10	2		R	An unsigned number specifying of active subcarriers.

B.5.4.11.11 StartingMinislot

When written by the CCAP Core this TLV specifies the future time (expressed as a 32-bit DOCSIS timestamp value) when the upstream channel change (signaled by MAP messages to CMs) goes into effect. The 32-bit timestamp points to the Alloc Start Time of the first MAP with incremented UCD Count. When read the RPD return zero.

[90] Revised per R-PHY-N-16.1556-3 on 8/10/16 by JB.

TLV Type	Length	Units	Access	Value
66.11	4		R/W	An unsigned 32-bit value specifying the future time expressed as a DOCSIS timestamp when the most recent channel configuration change will go into effect.

B.5.4.11.12 PreambleString

This TLV describes the This TLV defines the complete preamble pattern superstring including standard and extended preamble pattern strings, as defined for DOCSIS UCD message. The maximum length of the string is 128+64 = 192. The string is formatted conforming to the MULPI specification.

TLV Type	Length	Units	Access	Value
66.12	variable		R	The complete preamble pattern superstring including standard and extended preamble pattern string from UCD.

B.5.4.11.13 TargetRxPower

This attribute provides the power of the expected commanded received signal in the channel, referenced to the input of the RPD upstream RF port.

TLV Type	Length	Units	Access	Value
66.13	2	TenthdBmV	R/W	A signed number defining the commanded received signal in the channel.

B.5.4.11.14 EnableFlowTags

This TLV is used to instruct the RPD to insert Flow Tags value into UEPI headers on OFDM channels.

TLV Type	Length	Units	Access	Value
66.14	1		R/W	A Boolean value instructing the RPD to insert Flow Tags into UEPI headers. 0 - The RPD does not insert Flow Tags. 1 - The RPD inserts Flow Tags. Default value is 0.

B.5.4.11.15 ScramblerSeed

This TLV is used to configure upstream scrambler seed in the RPD. The rightmost 23 bits are used.

TLV Type	Length	Units	Access	Value
66.15	4		R	An unsigned integer value containing the scrambler seed in 23 LSB.

B.5.4.11.16 ConfigMultiSectionTimingMer[91]

This attribute allows the CCAP Core to configure the Multi-Section to subcarrier mapping for the selected channel's UEPI probe pseudowire the purpose of reporting timing error and MER. The number of sections is implied by the number of length divided by 32 bits. Each 32 bits is a pair of 16bit fields representing the low subcarrier (Lsc) and the high subcarrier (Hsc) index. The description of the UEPI pseudowire format can be found in [UEPI].

This configuration is based on the capability of the RPD.

TLV Type	Length	Units	Access	Value
66.16	variable	N/A	R/W	16bits for Lsc(1), 16bits for Hsc(1), …, 16bits for Lsc(M), 16 bits for Hsc(M)

[91] Added per R-PHY-N-16.1576-1 on 9/19/16 by JB.

B.5.4.11.17 BwReqAggrControlOfdma

This TLV is used to configure bandwidth request aggregation parameters for the selected OFDMA channel. The description of the bandwidth request aggregation function is provided in [R-UEPI]

TLV Type	Length	Units	Access	Value
66.17	variable		N/A	One or two sub-TLVs with bandwidth request aggregation control attributes.

B.5.4.11.18 MaxReqBlockEnqTimeout

This attribute is used to configure the maximum time a bandwidth request can be held in a queue on the RPD before the RPD sends it in a UEPI packet. This attribute controls bandwidth request aggregation for the selected OFDMA channel.

TLV Type	Length	Units	Access	Value
66.17.1	2	microseconds	N/A	An unsigned short value specifying the maximum time a bandwidth request can be held in a queue on the RPD. The valid range is 0 – 500 microseconds. The default value is 0.

B.5.4.11.19 MaxReqBlockEnqNumber

This attribute is used to configure the maximum number of bandwidth requests that the RPD can hold in a queue before the RPD sends them in a UEPI packet. This attribute controls bandwidth request aggregation for the selected OFDMA channel.

TLV Type	Length	Units	Access	Value
66.17.2	1		N/A	An unsigned byte value specifying the maximum number of bandwidth requests that the RPD can hold in a queue on the RPD. The valid range is 1 – 63. The default value is 1.

B.5.4.12 *Configuration of OFDMA Channel Initial Ranging IUC*

B.5.4.12.1 NumSubcarriers

This TLV defines the number of subcarriers for initial ranging (N_{ir}).

TLV Type	Length	Units	Access	Value
67.1	2		R	Number of subcarriers for initial ranging. Only even values are permitted.

B.5.4.12.2 Guardband

This TLV defines the number of guard subcarriers. It represents the sum of the upper and lower guard bands in Hz.

TLV Type	Length	Units	Access	Value
67.2	2		R	The number of guard subcarriers

B.5.4.13 *Configuration of OFDMA Channel Fine Ranging IUC*

B.5.4.13.1 NumSubcarriers

This TLV defines the number of subcarriers for fine ranging (N_{fr}).

TLV Type	Length	Units	Access	Value
68.1	2		R	Number of subcarriers for fine ranging. Only even values are permitted.

B.5.4.13.2 Guardband

This TLV defines the number of guard subcarriers. It is the sum of the upper and lower guard bands in Hz.

TLV Type	Length	Units	Access	Value
68.2	2		R	The number of guard subcarriers.

B.5.4.14 *Configuration of OFDMA Channel Data IUCs*

B.5.4.14.1 UsOfdmaDataIuc

UsOfdmaDataIuc represent a complex TLV used to report data IUC configuration of an OFDM channel.

TLV Type	Length	Units	Access	Value
69	variable		N/A	A sequence of sub-TLVs representing configuration of a single data IUC of an OFDM channel.

B.5.4.14.2 DataIuc

This TLV identifies the data IUC being configured.

TLV Type	Length	Units	Access	Value
69.1	1		N/A	An unsigned number identifying the data IUC being configured. The valid values are: 5, 6, 9, 10, 11, 12 and 13. All other values are reserved.

B.5.4.14.3 StartMinislotNum

This object represent the first minislot number in a range of minislots with a given configuration.

TLV Type	Length	Units	Access	Value
69.2	2		N/A	An unsigned 8-bit value specifying the first minislot number of a range of minislots with a given configuration. Valid number range is 0.,237.

B.5.4.14.4 FirstSubcarrierId

This TLV defines the first subcarrier where the first minislot starts.

TLV Type	Length	Units	Access	Value
69.3	2		R	An unsigned short specifying the first subcarrier of the minislot range.

B.5.4.14.5 NumConsecutiveMinislots

This TLV defines the number of consecutive minislots in a range.

TLV Type	Length	Units	Access	Value
69.4	2		R	An unsigned byte specifying the number of consecutive minislots with a given configuration.

B.5.4.14.6 MinislotPilotPattern

This TLV defines the pilot pattern for the minislot.

TLV Type	Length	Units	Access	Value
69.5	1		R	An unsigned byte specifying the pilot pattern for the minislot. The valid range for this object is {1..14}. This number corresponds to one of the pilot patterns defined in the [PHYv3.1] specification.

B.5.4.14.7 DataSymbolModulation

This TLV defines the modulation of the data symbols of a minislot.

TLV Type	Length	Units	Access	Value
69.6	1		R	An enumerated value defining the modulation of the data symbols of the minislots in the range. The valid values are: 1- other 2- zeroValued 3 - bpsk 4 - qam8 5 - qam16 6 - qam32 7 - qam64 8 - qam128 9 - qam256 10 - qam512 11 - qam1024 12 - qam2048 13 - qam4096

B.5.4.15 *Upstream OFDMA IUC Configuration TLVs*

The configuration of upstream OFDMA channel subcarriers are grouped in UsOfdmaSubcarrierCfg object. This object includes the following TLVs.

B.5.4.16 *Upstream OFDMA Subcarrier Usage Configuration*

B.5.4.16.1 UsOfdmaSubcarrierCfg

The configuration of the usage of subcarriers upstream OFDMA channel is grouped in UsOfdmaSubcarrierCfg complex TLV.

TLV Type	Length	Units	Access	Value
70	variable		N/A	A set of sub-TLVs representing a single range of subcarriers usage.

When the CCAP Core reads the upstream OFDM channel subcarrier usage configuration, the CCAP Core issues a read request with TLV 70. In response to such request the RPD returns a set of 69 TLVs, each TLV 69 representing a single range of configured minislots.

B.5.4.16.2 StartingSubcarrierId

This TLV defines the starting subcarrier number of a range.

TLV Type	Length	Units	Access	Value
70.1	2		R	An unsigned short number defining the subcarrier number of the first subcarrier in the range. Valid values are 0-4095.

B.5.4.16.3 NumConsecutiveSubcarriers

This TLV defines how many consecutive subcarriers are in the range.

TLV Type	Length	Units	Access	Value
70.2	2		R	An unsigned short value defining how many consecutive subcarriers are in the range.

B.5.4.16.4 SubcarrierUsage

This TLV specifies the subcarrier usage.

TLV Type	Length	Units	Access	Value
70.3	1		R	The subcarrier usage. The defined values are: 1 - Other 2 - Data 3 - Excluded 4 - Unused All other values are reserved.

B.5.5 Upstream QoS

DOCSIS 3.0 and later specifications define three types of upstream data bursts for SC-QAM channels. These are:

Legacy bursts are sent by DOCSIS 2.0 or older CMs and DOCSIS 3.0 CMs not operating in MTC mode.

Segment-header-on bursts are sent by DOCSIS 3.1 CMs and DOCSIS 3.0 CMs in MTC mode on Service Flows provisioned for segment-header-on operation.

Segment-header-off bursts are sent by DOCSIS 3.1 CMs and DOCSIS 3.0 CMs in MTC mode on Service Flows provisioned for segment-header-off operation.

DOCSIS 3.1 defines two types of upstream CCF segments for OFDMA channels: segment-header-on and segment-header-off.

In order to correctly decode incoming upstream transmissions, the upstream burst receiver in the RPD needs to be configured by the CCAP Core with the type of upstream data burst or CCF segment type that CMs can send on a particular data SID.

The SidQos TLV permits the CCAP Core to configure three attributes for a range of SIDs in the RPD:

- Service Flow Type, indicating the type of upstream bursts/CCF segment.

- UEPI PSP Flow Id, the identifier of UEPI PSP flow associated with a SID.

- UEPI Flow Tag value.

The encoding of the SidQos TLV allows writing and reading of these attributes for individual SIDs or for ranges of SIDs. While the protocol allows issuing SidQos TLV for a large SID ranges, this may not always be possible because the SID attributes may not be uniform for large SID ranges. There are special considerations for reading the SidQos attributes. When the CCAP Core reads SidQos attributes for a range of SIDs, the RPD's response can include a larger number of TLVs as necessary to convey the current variability of SidQos attributes. The CCAP Core needs to ensure that the SIDs range is sufficiently small to allow the RPD to respond with a set of TLVs that fit into a single RCP message.

The UML model for SidQos is shown in Figure 34.

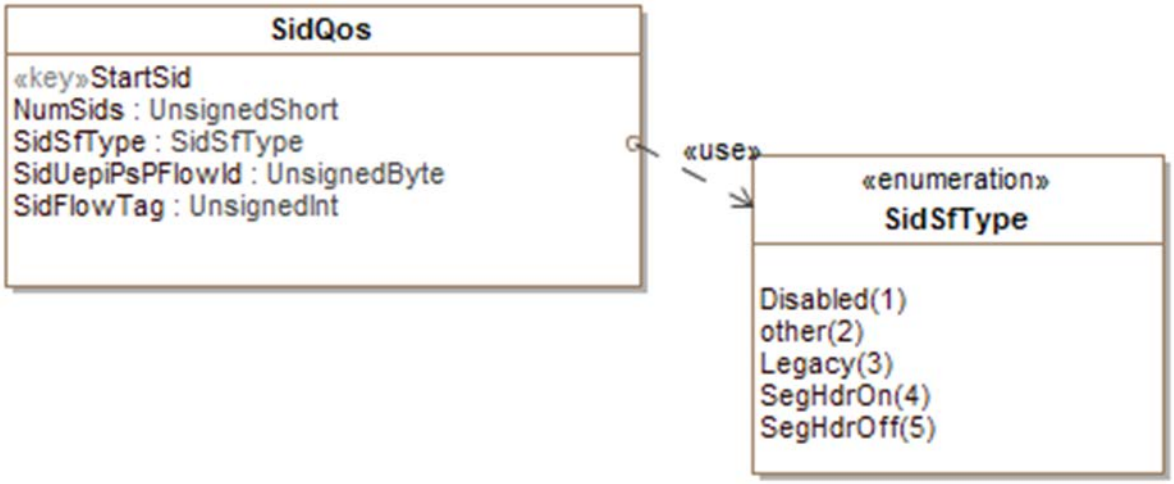

Figure 34 - SidQos Configuration Objects

B.5.5.1 *SidQos*

SidQos is a complex TLV used to configure RPD's upstream SID attributes for a particular upstream channel.

TLV Type	Length	Units	Access	Value
96	Variable			A set of sub-TLVs defining a single range of SIDs for whom the service flow type and/or UEPI priority is configured.

B.5.5.2 *StartSid*

StartSid is a TLV used to select the first SID in a range configured by SidQos TLV. The sender MUST include a single StartSid TLV within SidQos TLV as the first sub-TLV.

TLV Type	Length	Units	Access	Value
96.1	2		N/A	An unsigned short number defining the first SID in the selected range. Valid values are 1 - 15871.

B.5.5.3 *NumSids*

NumSid is a TLV used to select the number of SIDs in a range configured by SidQos TLV. The sender MUST include a single NumSids TLV within SidQos TLV as the second sub-TLV.

TLV Type	Length	Units	Access	Value
96.2	2		N/A	An unsigned short number defining the number of SIDs in the selected range. Valid values are 1 - 15871.

B.5.5.4 *SidSfType*

SidSfType is a TLV which the type of upstream transmission for a range of SIDs.

TLV Type	Length	Units	Access	Value
96.3	1		R/W	The type of upstream transmission/CCF segment. 0 - SID is disabled. 1 - Other. 2 - Legacy, this value is valid for SC-QAM channels. 3 - Segment-header-on. 4 - Segment-header-off. 0 is the default value.

B.5.5.5 *SidUepiFlowId*

SidUepiFlowId is a TLV used to configure the RPD to forward data received on a SID on the selected PSP flow.

TLV Type	Length	Units	Access	Value
96.4	1		R/W	An unsigned byte number identifying the PSP flow number to be used for data received on a SID. Valid values are 0-8. Default value is 0.

B.5.5.6 *SidFlowTag*

SidFlowTag is a TLV used to program Flow Tag value for a range of SIDs.

TLV Type	Length	Units	Access	Value
96.5	4		R/W	An unsigned integer number specifying the Flow Tag value. Default value is 0.

B.5.5.7 *SidQos TLV example*

The example shown below represents a REX Write request, in which the CCAP Core sets the SidQos values for ATDMA channels 2 and 3 on upstream RF port 6. SIDs from 1 to 0x1fff are configured for segment-header-on operation and assigned to UEPI PSP flow 1, while SIDs from 0x2000 to 0x2fff are configured for segment-header-off operation and UEPI PSP flow 0.

```
{ T = REX, L = 132, V =                      ; top-level “container” type
   { T = Sequence, L = 129, V =                  ; a seq. of TLVs starting with oper.
      { T = SequenceNumber, L = 2, V = 21 }
      { T = Operation, L = 1, V = Write }
      { T = RfChannel, L = 57, V =           ; L = 15+21+21 = 57
         { T = RfChannelSelector, L = 12, V =
            { T = RfPortIndex, L = 1, V = 6
            { T = RfChannelType, L = 1, V = 6 } ; ATDMA channel
            { T = RfChannelIndex, L = 1, V = 2}
         }
         { T = SidQos, L = 18, V =
            { T = StartSid, L = 2, V = 1 }
            { T = NumSids, L = 2, V = 0x1fff}
            { T = SidSfType, L = 1, V = 3}       ;segment-header-on
            { T = SidUepiFlowId, L = 1, V = 1}   ;selected PSP flow 1
         }
         { T = SidQos, L = 18, V =
            { T = StartSid, L = 2, V = 0x2000}
            { T = NumSids, L = 2, V = 0x0fff}
            { T = SidSfType, L = 1, V = 4}       ;segment-header-off
            { T = SidUepiFlowId, L = 1, V = 1}   ;selected PSP flow 1
         }
      }
      { T = RfChannel, L =  57, V =
         { T = RfChannelSelector, L = 12, V =
            { T = RfPortIndex, L = 1, V = 2 }
```

```
            { T = RfChannelType, L = 1, V = 16 } ; ATDMA channel
            { T = RfChannelIndex, L = 1, V = 3}
         }
         { T = SidQos, L = 18, V =
            { T = StartSid, L = 2, V = 1 }
            { T = NumSids, L = 2, V = 0x1fff}
            { T = SidSfType, L = 1, V = 3}       ;segment-header-on
            { T = SidUepiFlowId, L = 4, V = 0x1}   ;selected PSP flow 1
         }
         { T = SidQos, L = 18, V =
            { T = StartSid, L = 2, V = 0x2000}
            { T = NumSids, L = 2, V = 0x0fff}
            { T = SidSfType, L = 1, V = 4}       ;segment-header-on
            { T = SidUepiFlowId, L = 1, V = 0}   ;selected PSP flow 0
         }
      }
   }
}
```

B.5.5.8 *UsRfPort*[92]

This complex TLV specifies is used to communicate configuration information related to upstream RF port.

TLV Type	Length	Units	Access	Value
98	variable		N/A	One or more sub-TLVs.

B.5.5.8.1 AdminState

This attribute configures the administrative state for the US RF Port.

TLV Type	Length	Units	Access	Value
98.1	1		R/W	The administrative state of the upstream RF Port. The defined values are: 1 - “other” 2 - “up” 3 - “down” 4 - “testing” Values 0, 5-255 are reserved.

B.5.5.8.2 BwReqAggrControl

This TLV is used to configure bandwidth request aggregation parameters for all SC-QAM channels associated with an US RF port on the RPD. The description of the bandwidth request aggregation function is provided in [R-UEPI]

TLV Type	Length	Units	Access	Value
98.2	variable		N/A	One or two sub-TLVs with bandwidth request aggregation control attributes.

B.5.5.8.3 MaxReqBlockEnqTimeout

This attribute is used to configure the maximum time a bandwidth request can be held in a queue on the RPD before the RPD sends it in a UEPI packet. This attribute controls bandwidth request aggregation for all SC-QAM channels associated with an US RF port.

[92] Revised per R-PHY-n-16.1644-3 on 12/20/16 by JB.

TLV Type	Length	Units	Access	Value
98.2.1	2	microseconds	N/A	An unsigned short value specifying the maximum time a bandwidth request can be held in a queue on the RPD. The valid range is 0 – 500 microseconds. The default value is 0.

B.5.5.8.4 MaxReqBlockEnqNumber

This attribute is used to configure the maximum number of bandwidth requests that the RPD can hold in a queue before the RPD sends them in a UEPI packet. This attribute controls bandwidth request aggregation for all SC-QAM channels associated with an US RF port.

TLV Type	Length	Units	Access	Value
98.2.2	1		N/A	An unsigned byte value specifying the maximum number of bandwidth requests that the RPD can hold in a queue on the RPD. The valid range is 1 – 63. The default value is 1.

B.5.6 Device Management TLVs

The set of RCP objects used in device management is presented in Figure 35.

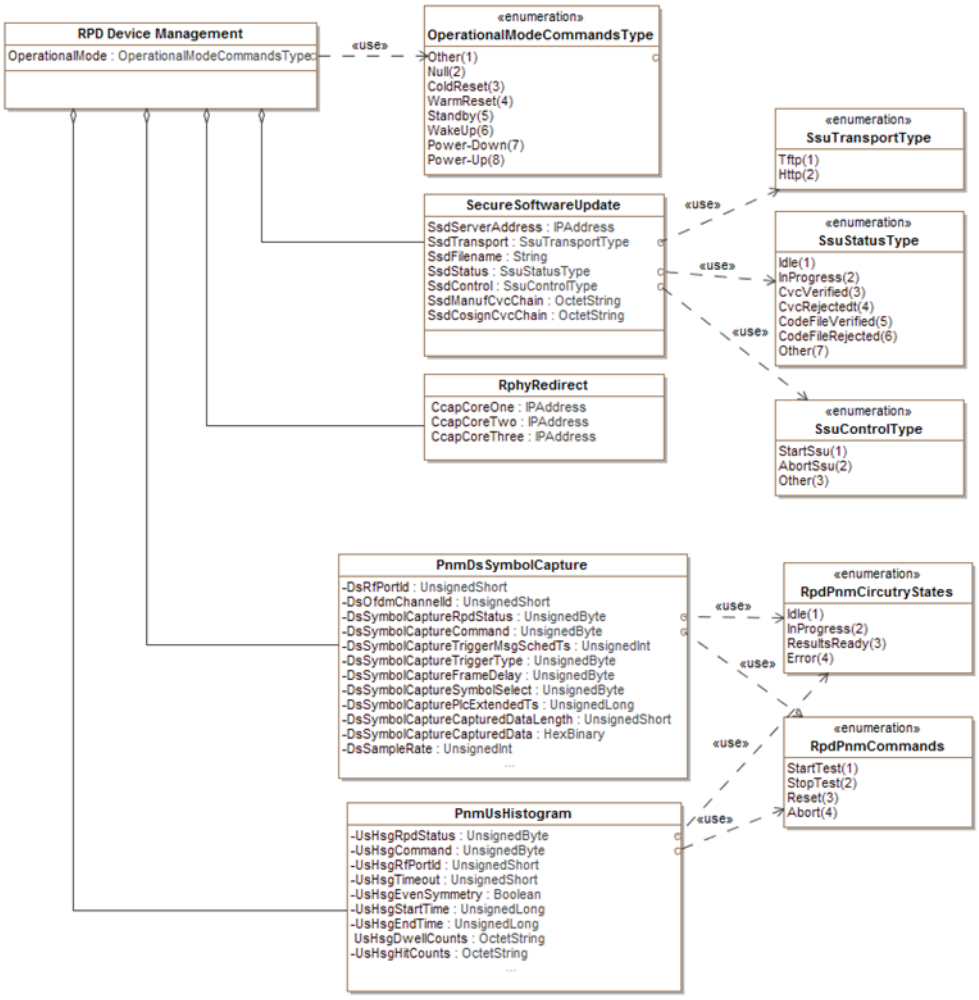

Figure 35 - RCP Device Management Objects

B.5.6.1 *Secure Software Download*

This complex TLV is used to communicate parameters and status of secure software download.

TLV Type	Length	Units	Access	Value
90	variable		R/W	A set of TLVs with parameters or status information related to Secure Software Download.

B.5.6.1.1 SSD Server Address

This TLV conveys the IP address of SSD Server.

TLV Type	Length	Units	Access	Value
90.1	4 or 16		R/W	The IP address of the SSD server. A length of 4 indicated IPv4 address. A length of 16 indicated IPv6 address.

B.5.6.1.2 SSD Transport

The SSD Transport TLV is used to communicate the type of transport for the RPD download of the software file.

TLV Type	Length	Units	Access	Value
90.2	1		R/W	The defined values are: 1 - TFTP 2 - HTTP All other values are reserved.

B.5.6.1.3 SSD Filename

The SSD Filename TLV is used to communicate the name of the software file which the RPD needs to download.

TLV Type	Length	Units	Access	Value
90.3	variable		R/W	String containing the ASCI filename of the file which the RPD needs to download.

B.5.6.1.4 SSD Status

The SSD Status TLV allows the CCAP Core to read the status of the SSD process in the RPD.

TLV Type	Length	Units	Access	Value
90.4	1		RO	An unsigned byte with one of the defined values: 1- Other 2 - Idle 3 - In progress 4 - CvcVerified 5 - CvcRejected 6 - CodeFileVerified 7 - CodeFileRejected All other values are reserved.

B.5.6.1.5 SSD Control

The SSD Control TLV allows the CCAP Core to maintain control over the SSD process. When read, this object returns the latest value written to it.

TLV Type	Length	Units	Access	Value
90.5	1		R/W	An unsigned byte with one of the defined values: 1 - Other 2 - StartSsu 3 - AbortSsu All other values are reserved.

B.5.6.1.6 SSD Manufacturer CVC Chain

The certificate chain from the new PKI that contains both the Manufacturer Code Verification Certificate and the certification authority (CA) certificate that issued the Manufacturer Code Verification Certificate for Secure

Software Download. The Manufacturer CVC Chain TLV (M-CVC-C) is used to enable the RPD to download the code file from the download server.

TLV Type	Length	Units	Access	Value
90.6	variable		R/W	An octet string with Manufacturer CVC Chain (degenerate PKCS7 signedData structure that contains the CVC and the CVC CA certificate chain from the new PKI in the certificates field).

B.5.6.1.7 Co-signer CVC Chain[93]

The certificate chain from the new PKI that contains both the Co-signer Code Verification Certificate and the certification authority (CA) certificate that issued the Co-signer Code Verification Certificate for Secure

Software Download. The Co-signer CVC Chain TLV (C-CVC-C) is used to enable the RPD to download the code file from the download server.

TLV Type	Length	Units	Access	Value
90.7	variable		R/W	An octet string with Co-signer CVC Chain Certificate (degenerate PKCS7 signedData structure that contains the CVC and the CVC CA certificate chain from the new PKI in the certificates field).

B.5.7 OOB SCTE 55-1 Configuration TLVs[94]

The UML model of RPD's SCTE 55-1 configuration is shown on Figure 36 and Figure 37. The RPD supports only one downstream 55-1 channel per downstream RF port. In addition to standard attributes (AdminState, CcapCoreOwner, RfMute) there are only two other attributes defined for the downstream channel: Frequency and PowerAdjust. The RPD can operate up to three upstream STCE 55-1 channels for each upstream RF port. There is only one specific attribute defined for the upstream channel: Frequency and three attributes necessary to identify data within the virtual ARPD. The upstream 55-1 demodulator operates with a constant reference power level of 0 dBmV. The CCAP Core cannot configure upstream power reference level for 55-1 demodulator.

[93] Removed sections B.6.5.1 through B.7.15.3 per R-PHY-N-16.1444-1, and added B.6.7 through B.6.8.20 per R-PHY-N-16.1451-1 on 4/15/16 by JB.
[94] Added per R-PHY-N-16.1562-1 on 8/10/16 by JB.

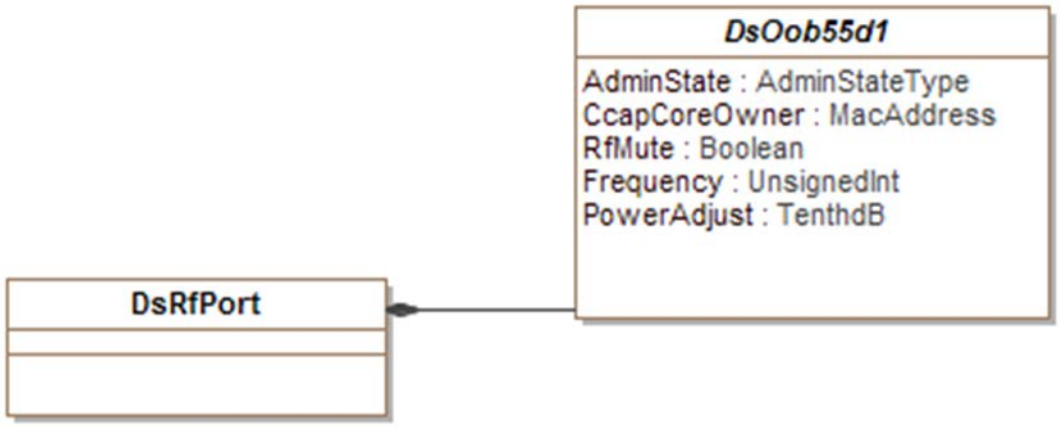

Figure 36 - SCTE 55-1 Downstream Channel Configuration

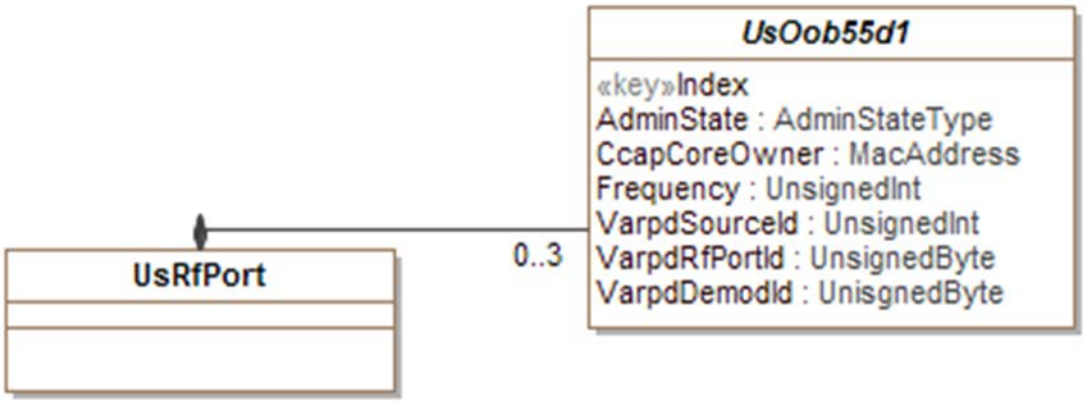

Figure 37 - SCTE 55-1 Upstream Channel Configuration

B.5.7.1 *DsOob55d1*

The DsOob55d1 is a complex TLV used to communicate attributes related to configuration of the downstream SCTE 55-1 OOB channels in the RPD.

TLV Type	Length	Units	Access	Value
91	variable		N/A	A sequence of sub-TLVs representing configuration of RPD's downstream SCTE 55-1 out-of-band channel.

B.5.7.1.1 AdminState

This attribute describes the administrative state of the SCTE 55-1 channel.

TLV Type	Length	Units	Access	Value
91.1	1		R/W	The administrative state of the SCTE 55-1 downstream channel. The AdminState can have possible values: 1 - “other” 2 - “up” 3 - “down” 4 - “testing” Values 0, 5-255 are reserved.

B.5.7.1.2 CcapCoreOwner

This TLV specifies the identifier of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
91.2	6		R/W	The MAC address of the CCAP Core operating the channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.7.1.3 RfMute

This attribute permits the CCAP Core to configure the mute state of the SCTE 55-1 downstream channel. If set to true, the channel is in the muted state.

TLV Type	Length	Units	Access	Value
91.3	1		R/W	A Boolean value which specifies whether the selected channel is in the muted state. 0 - Channel is not muted. 1 - Channel is muted. Values 2-255 are reserved.

B.5.7.1.4 Frequency

This attribute specifies the center frequency of the SCTE 55-1 downstream channel in Hertz.

TLV Type	Length	Units	Access	Value
91.4	4	Hertz	R/W	Center frequency of the SCTE 55-1 downstream channel specified in units of Hertz. The value can range 71 to 129 MHz in steps of 50kHz. The default value is 75250000 Hz.

The CCAP Core MUST support configuration of the frequency of SCTE 55-1 downstream channel in range of 71 MHz to 129 MHz in steps of 50 kHz.

NOTE: The frequency range defined above is narrower than the 70-130 MHz range defined in SCTE 55-1 because of the practical limitations of the deployed STB devices and downstream modulators.

B.5.7.1.5 PowerAdjust

This object specifies power level adjustment for the SCTE 55-1 downstream channel relative to the base power level configured for the corresponding DS RF port.

TLV Type	Length	Units	Access	Value
91.5	1	TenthdB	R/W	A signed value of power level adjustment amount in units of 0.1 dB relative to the base power level specified for the corresponding DS RF port.

B.5.7.2 *UsOob55d1*

The UsOob55d1 is a complex TLV used to communicate attributes related to configuration of the upstream SCTE 55-1 OOB channels in the RPD.

TLV Type	Length	Units	Access	Value
92	Variable		N/A	A sequence of sub-TLVs representing configuration attributes of RPD's upstream SCTE 55-1 out-of-band channel.

B.5.7.2.1 AdminState

This attribute communicates the administrative state of the SCTE 55-1 upstream channel.

TLV Type	Length	Units	Access	Value
92.1	1		R/W	The administrative state of the SCTE 55-1 upstream channel. The AdminState can have possible values: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.7.2.2 CcapCoreOwner

This attribute specifies the identifier of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
92.2	6		R/W	The MAC address of the CCAP Core operating the channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.7.2.3 Frequency

This attribute specifies the center frequency of the SCTE 55-1 upstream channel in Hertz.

TLV Type	Length	Units	Access	Value
92.3	4	Hertz	R/W	Center frequency of the SCTE 55-1 upstream channel specified in units of Hertz. Range 8.096 MHz to 40.160 MHz in 192 kHz steps.

The CCAP Core MUST support configuration of the frequency of SCTE 55-1 upstream channel in range of 8.096 MHz to 40.160 MHz in steps of 192 kHz.

B.5.7.2.4 VarpdDeviceId

This attribute specifies the identifier used in virtual ARPD protocol.

TLV Type	Length	Units	Access	Value
92.4	4	N/A	R/W	A 32-bit identifier used in virtual ARPD protocol. There are no defined restrictions on the value of this attribute.

B.5.7.2.5 VarpdRfPortId

This attribute specifies the RF Port identifier which is used in virtual ARPD protocol.

TLV Type	Length	Units	Access	Value
92.5	1	N/A	R/W	An 8-bit identifier of the RF port used in virtual ARPD protocol. There are no defined restrictions on the value of this attribute.

B.5.7.2.6 VarpdDemodId

This attribute specifies the Demodulator identifier which is used in virtual ARPD protocol.

TLV Type	Length	Units	Access	Value
92.6	1	N/A	R/W	An 8-bit identifier of the demodulator used in virtual ARPD protocol. The valid range of values is 0..23. All other values are reserved.

B.5.8 OOB SCTE 55-2 Configuration TLVs

The UML model of RPD's SCTE 55-2 is shown in Figure 38. An RPD can incorporate a number of SCTE 55-2 modules, each represented by Oob55d2Module object. The number of SCTE 55-2 Modules is communicated via RPD capabilities. Common parameters for all 55-2 modules are grouped into an Oob55-2Config object. Each SCTE55-2 module consists of exactly one modulator and between one and eight demodulators, represented by Oob55d2Modulator and Oob55d2Demodulator objects, respectively. Oob55-2Modulator can be associated with one or more downstream RF ports. Oob55-2Demodulator can be associated with zero or one upstream RF ports. The RPD reports these association to the CCAP Core through DsPortAssociation and UsPortAssociation read-only objects. This specification does provide a method for configuring these associations.

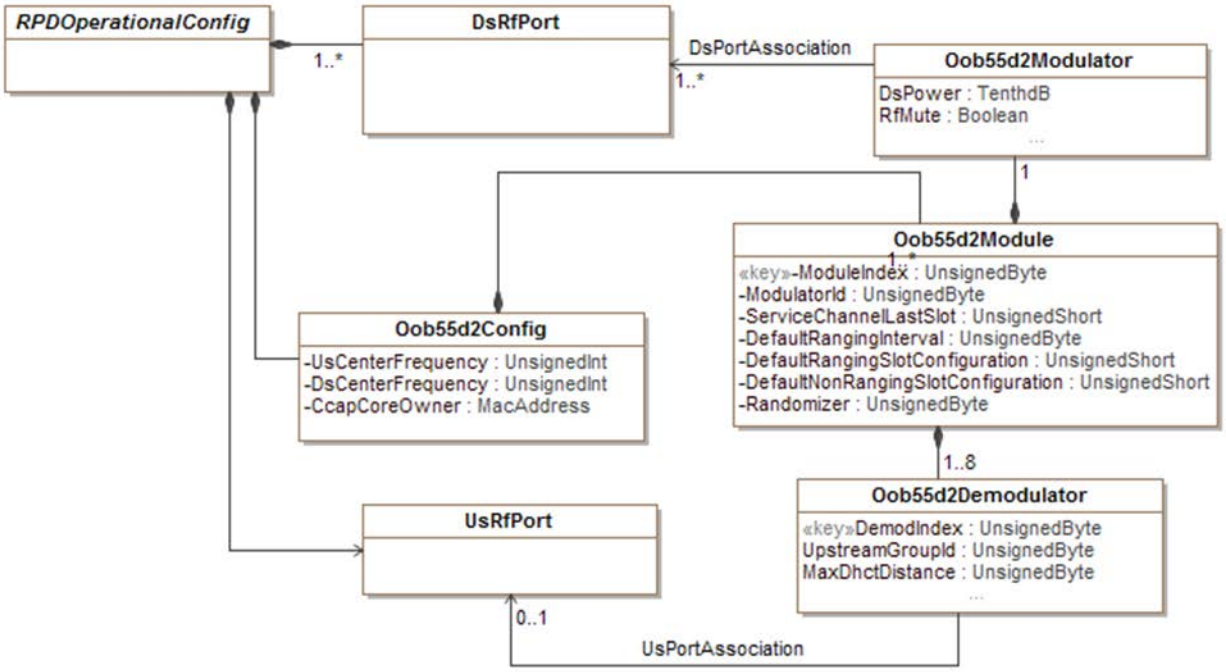

Figure 38 - SCTE 55-2 OOB Configuration Objects

B.5.8.1 *Oob55d2Config*

The Oob55d2Config is a complex TLV used to communicate attributes related to configuration of SCTE 55-2 OOB functions in the RPD.

TLV Type	Length	Units	Access	Value
93	variable		R/W	A sequence of sub-TLVs representing configuration of RPD's SCTE 55-2 out-of-band functions.

B.5.8.2 *DsCenterFrequency*

The DsCenterFrequency TLV is used to configure the center frequency common for all SCTE 55-2 OOB modulators in the RPD.

TLV Type	Length	Units	Access	Value
93.1	4	Hertz	R/W	The center frequency for all SCTE 55-2 downstream modulators in the RPD.

B.5.8.3 *UsCenterFrequency*

The UsCenterFrequency TLV is used to configure the center frequency for all SCTE 55-2 demodulators in the RPD.

TLV Type	Length	Units	Access	Value
93.2	4	Hertz	R/W	The center frequency for all SCTE 55-2 upstream demodulators in the RPD.

B.5.8.4 *CcapCoreOwner*

This TLV specifies the MAC address of the CCAP Core which operates the SCTE 55-2 OOB functions in the RPD. This TLV can be present only one time in the Oob55d2Config TLV.

TLV Type	Length	Units	Access	Value
93.3	4	Hertz	R/W	The MAC address of the CCAP Core operating the SCTE 55-2. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.8.5 *Oob55d2Module*

Oob55d2Module is a complex TLV used to communicate configuration attributes of a single SCTE 55-2 OOB module. This TLV can be present multiple times in the Oob55d2Config TLV, once for each configured 55-2 module.

TLV Type	Length	Units	Access	Value
93.4	variable			A set of sub-TLVs representing configuration of a single SCTE 55-2 OOB module.

B.5.8.6 *ModuleIndex*

ModuleIndex TLV carries the index of a SCTE 55-2 module in Oob55d2Module TLV. This TLV is required to be present exactly once and to be the first sub-TLV inside the Oob55d2Module TLV.

TLV Type	Length	Units	Access	Value
93.4.1	1		N/A	A zero based index of the SCTE 55-2 module in the RPD. The valid range is from 0 to NumOob55d2Modules - 1.

B.5.8.7 *ModulatorId*

ModulatorId TLV carries the protocol identifier of a SCTE 55-2 module in Oob55d2Module TLV.

TLV Type	Length	Units	Access	Value
93.4.2	1		R/W	An identifier of the modulator in the RPD. This value is included in all SCTE 55-2 upstream packets so as to identify which modulator the packets came from.

B.5.8.8 *ServiceChannelLastSlot*

ServiceChannelLastSlot TLV is used to configure the maximum value of the ESF counter.

TLV Type	Length	Units	Access	Value
93.4.3	2		R/W	Maximum value of the ESF counter (10 bits) before it rolls over. The default value is 0x3E8.

B.5.8.9 *DefaultRangingInterval*

DefaultRangingInterval TLV is used to configure the frequency of use of the Default Ranging Slot Configuration values when using the default slot allocation for DAVIC frame generation.

TLV Type	Length	Units	Access	Value
93.4.4	1		R/W	The frequency of use of the Default Ranging Slot Configuration values when using the default slot allocation for DAVIC frame generation. 0 - never 1 - every frame, 2 - every 2 frames, etc. The default value is 8.

B.5.8.10 *DefaultRangingSlotConfiguration*

DefaultRangingSlotConfiguration TLV is used to define the slot configuration to output whenever the default slot allocation generator outputs an allocation with a ranging slot.

TLV Type	Length	Units	Access	Value
93.4.5	2		R/W	A 9-bit value defining the slot configuration to output whenever the default slot allocation generator outputs an allocation with a ranging slot. The default value is 0x10.

B.5.8.11 *DefaultNonRangingSlotConfiguration*

DefaultNonRangingSlotConfiguration TLV is used to define the slot configuration to output whenever the default slot allocation generator outputs an allocation without a ranging slot.

TLV Type	Length	Units	Access	Value
93.4.6	2		R/W	A 9-bit value defining the slot configuration to output whenever the default slot allocation generator outputs an allocation without a ranging slot. The default value is 0x1B.

B.5.8.12 *Randomizer*

Randomizer TLV is used to select a polynomial for the randomizer.

TLV Type	Length	Units	Access	Value
93.4.7	1		R/W	A polynomial selection value. 0 - corresponds to polynomial X6+X+1. 1 - corresponds to polynomial X6+X5+1. All other values are reserved. The default value is 0.

B.5.8.13 *DsPower*

DsPower TLV is used to configure the power level of the downstream modulator.

TLV Type	Length	Units	Access	Value
93.4.8	1	TenthdB	R/W	Power level of downstream modulator, in TenthdB relative to QAM256 power level.

B.5.8.14 *DsPortAssociation*

DsPortAssociation TLV communicates the association of the SCTE 55-2 modulator with RPD's downstream RF ports.

TLV Type	Length	Units	Access	Value
93.4.9	variable		RO	A list of indexes of associated DS RF ports. A zero length value field denotes that the modulator is not associated with any DS RF port.

B.5.8.15 *Oob55d2Demod*

Oob55d2Demod is a complex TLV used to communicate configuration attributes of a single SCTE 55-2 OOB demodulator. This TLV can be present multiple times in the Oob55d2Module TLV, once for each configured 55-2 demodulator.

TLV Type	Length	Units	Access	Value
93.4.10	variable			A set of sub-TLVs representing configuration of a single SCTE 55-2 OOB demodulator.

B.5.8.16 *DemodIndex*

DemodIndex TLV carries the index of a SCTE 55-2 demodulator with 55-2 module. This TLV is required to present exactly once and to be the first sub-TLV inside the Oob55d2Demod TLV.

TLV Type	Length	Units	Access	Value
93.4.10.1	1			A zero based identifier of the demodulator in the SCTE 55-2 module.

B.5.8.17 *UpstreamGroupId*

UpstreamGroupId TLV is used the configure identifier of a SCTE 55-2 demodulator in Oob55d2demod TLV.

TLV Type	Length	Units	Access	Value
93.4.10.2	1		R/W	A zero based protocol identifier of the demodulator in a SCTE 55-2 module. Valid range is 0-7. The default value is the same as corresponding DemodIndex.

B.5.8.18 *MaxDhctDistance*

MaxDhctDistanceTLV carries the identifier of a SCTE 55-2 demodulator in Oob55d2demod TLV. This TLV is required to be present exactly once and to be the first sub-TLV inside the Oob55d2Demod TLV.

TLV Type	Length	Units	Access	Value
93.4.10.3	1	31 km	R/W	The distance from the RPD to the furthest DHCT in units of 31km. Range 0- 8 corresponding to 0 km - 248 km; This value is converted into a timing offset in the RPD to apply to incoming cell receive times. The default value is 0.

B.5.8.19 *UsPortAssociation*

UsPortAssociation TLV communicates the association of the SCTE 55-2 demodulator with RPD's upstream RF port.

TLV Type	Length	Units	Access	Value
93.4.10.4	0 | 1		RO	An index of associated US RF port. A zero length value field denotes that the demodulator is not associated with any US RF port.

B.5.8.20 *RfMute*

RfMute TLV is used to mute the 55-2 modulator output. If set to true, the modulator is in the muted diagnostic state i.e., transmitting no signal.

TLV Type	Length	Units	Access	Value
93.4.11	1	N/A	R/W	A Boolean value which specifies whether the selected modulator is in the muted state. 0 - Modulator is not muted. 1 - Modulator is muted. Values 2-255 are reserved.

B.5.9 NDF Configuration TLVs

The UML model of RPD's NDF configuration attributes is shown in Figure 39. An RPD can support a number of NDF channels, each represented by NdfConfig object. The RPD communicates the number of supported NDF channels via RPD capabilities.

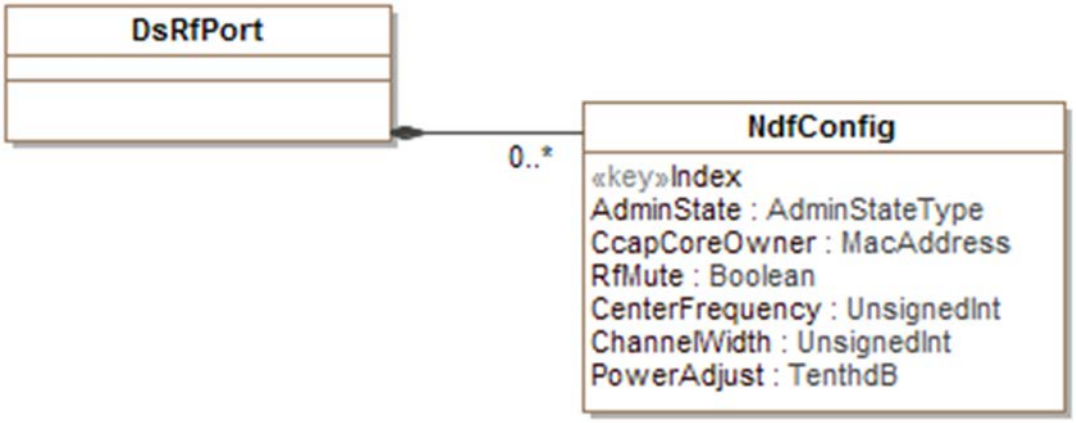

Figure 39 - NDF Configuration Objects

B.5.9.1 *NdfConfig*

The NdfConfig is a complex TLV used to communicate attributes related to configuration of a selected NDF channel in the RPD.

TLV Type	Length	Units	Access	Value
94	variable		R/W	A sequence of sub-TLVs representing configuration of RPD's NDF channel.

B.5.9.1.1 AdminState

This TLV describes the administrative state for the selected NDF channel.

TLV Type	Length	Units	Access	Value
94.1	1		R/W	The administrative state of the NDF Channel. The following values of the AdminState have been defined: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.9.1.2 CcapCoreOwner

This TLV specifies the MAC address of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
94.2	6	MacAddress	R/W	The MAC address of the CCAP Core operating the NDF channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.9.1.3 RfMute

RfMute TLV is used to mute the NDR modulator output. If set to true, the modulator is in the muted diagnostic state i.e., transmitting no signal.

TLV Type	Length	Units	Access	Value
94.3	1	N/A	R/W	A Boolean value which specifies whether the selected channel's modulator is in the muted state. 0 - Channel is not muted. 1 - Channel is muted. Values 2-255 are reserved.

B.5.9.1.4 CenterFrequency

This TLV specifies the center frequency of the channel in Hz.

TLV Type	Length	Units	Access	Value
94.4	4	Hertz	R/W	The center frequency of the NDF channel specified in unit of Hertz.

B.5.9.1.5 ChannelWidth

This TLV specifies the width of the NDF channel

TLV Type	Length	Units	Access	Value
94.5	4	Hertz	R/W	The width of the NDF channel in units of Hertz.

B.5.9.1.6 PowerAdjust

This object specifies power level adjustment for the NDF channel relative to the base power level configured for the DS RF port.

TLV Type	Length	Units	Access	Value
94.6	1	TenthdB	R/W	A signed value of power level adjustment amount in units of 0.1 dB relative to the base power level specified for the DS RF port.

B.5.10 NDR Configuration TLVs

The UML model of RPD's NDR configuration attributes is shown in Figure 40. An RPD can support a number of NDR channels, each represented by NdrConfig object. The RPD communicates the number of supported NDR channels via RPD capabilities.

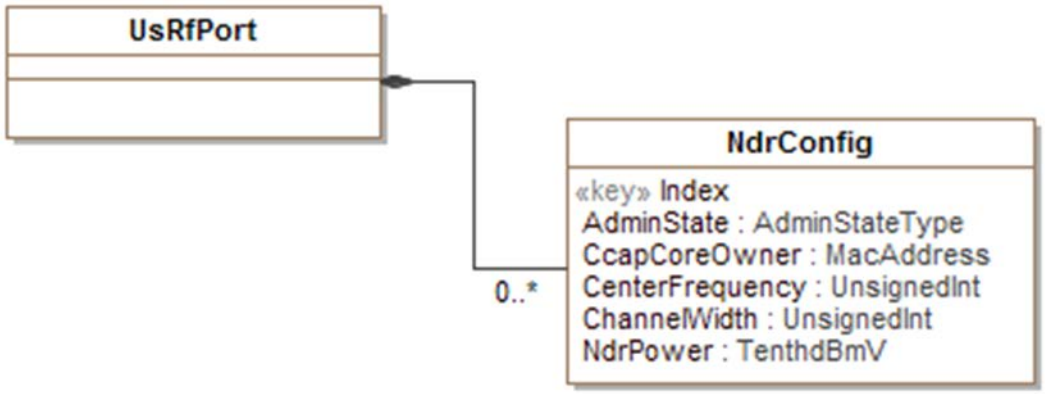

Figure 40 - NDR Configuration Objects.

B.5.10.1 *NdrConfig*

The NdrConfig is a complex TLV used to communicate attributes related to configuration of a selected NDR channel in the RPD.

TLV Type	Length	Units	Access	Value
95	variable		R/W	A sequence of sub-TLVs representing configuration of RPD's NDR channel.

B.5.10.1.1 AdminState

This TLV describes the administrative state for the selected NDR channel.

TLV Type	Length	Units	Access	Value
95.1	1		R/W	The administrative state of the NDR channel. The following values of the AdminState have been defined: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved.

B.5.10.1.2 CcapCoreOwner

This TLV specifies the MAC address of the CCAP Core which operates the channel.

TLV Type	Length	Units	Access	Value
95.2	6	MacAddress	R/W	The MAC address of the CCAP Core operating the NDR channel. When no CCAP Core operates the channel the RPD reports a NULL MAC address.

B.5.10.1.3 CenterFrequency

This TLV specifies the center frequency of the NDR channel in Hz.

TLV Type	Length	Units	Access	Value
95.3	4	Hertz	R/W	Center frequency of the NDR channel specified in units of Hertz.

B.5.10.1.4 ChannelWidth

This TLV specifies the width of the NDR channel.

TLV Type	Length	Units	Access	Value
95.4	4	Hertz	R/W	The width of the NDR channel in Hertz.

B.5.10.1.5 NdrPower

This object specifies the commanded input power level for the NDR channel as specified in section 7.3 of [R-OOB].

TLV Type	Length	Units	Access	Value
95.5	1	TenthdBmV	R/W	A signed value representing the commanded input power level in units of 0.1 dBmV.

B.5.11 RDTI Configuration TLVs[95]

The UML model of RPD's RDTI slave configuration attributes is shown on Figure 41. The configuration attributes have been divided into common attributes (TLVs 97.1 - 97.7) and per PTP port attributes represented by RpdPtpPortConfig (97.8) TLV. An RPD can support a number of PTP ports per CIN facing Ethernet port. The RPD communicates the number of supported PTP ports per Ethernet port via RPD capabilities.

[95] Revised per R-PHY-N-16.1564-2 on 8/22/16 by JB.

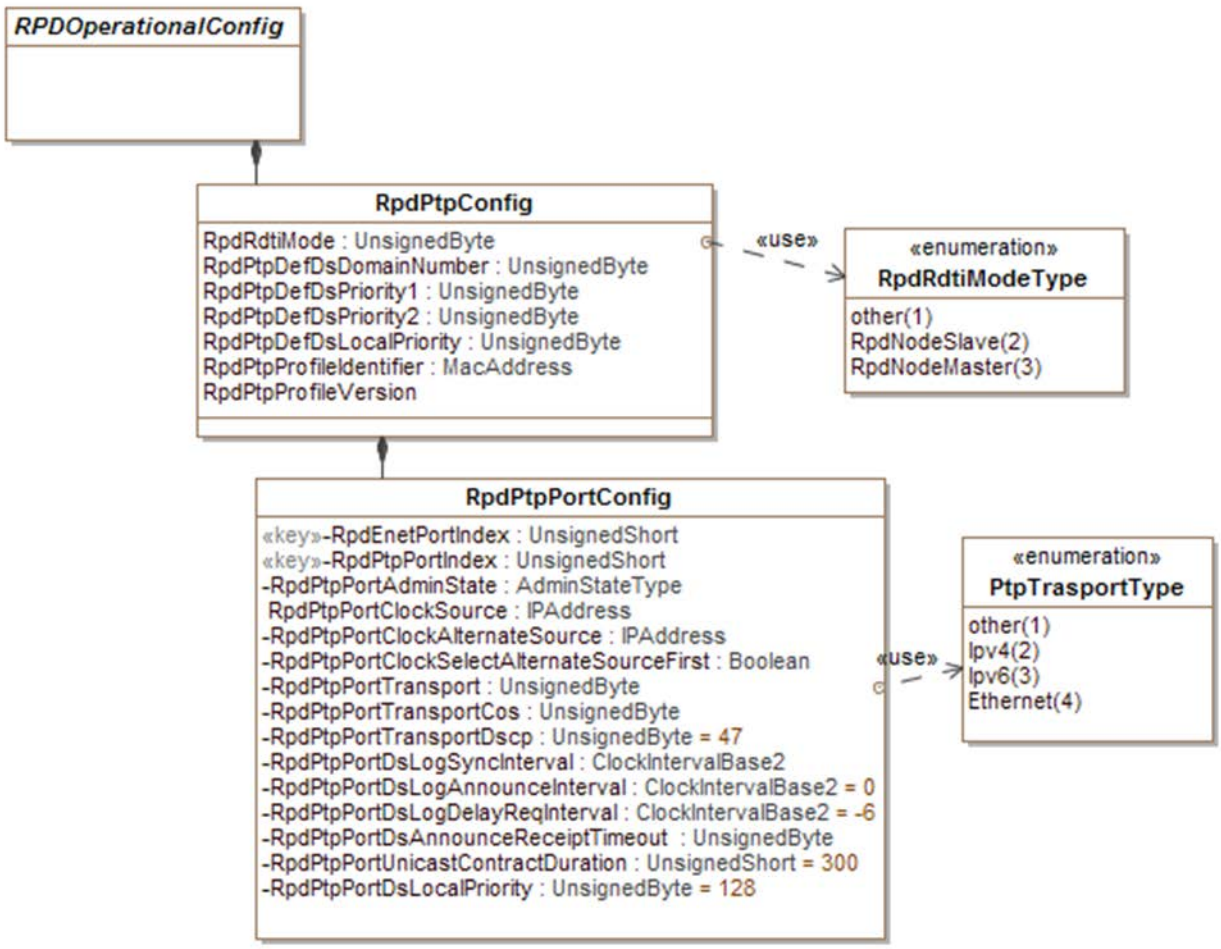

Figure 41 - RPD RDTI Configuration Attributes

B.5.11.1 *RdtiConfig*

RdtiConfig is a complex TLV used to communicate attributes related to configuration of the RDTI client in the RPD.

TLV Type	Length	Units	Access	Value
97	variable		N/A	A sequence of sub-TLVs representing configuration attributes of RPD's RDTI client.

B.5.11.1.1 RpdRdtiMode

RpdRdtiMode is a TLV used to configure the operational mode of the RDTI client in the RPD. This attribute does not have an equivalent attribute in IEEE 1588 specification or ITU G.8275.2. The RPD supports a single RDTI Mode applicable to DOCSIS, MPEG Video, Precision and NDF & NDR timing synchronization.

TLV Type	Length	Units	Access	Value
97.1	1		R/W	An unsigned byte value representing the operational mode of the RDTI client in the RPD. The following values of the RpdRdtiMode attribute have been defined: 1 - "other" 2 - "RpdNodeSlave", which correspond to Node_Slave mode defined in [R-DTI]. 3 - "RpdNodeMaster", which corresponds to Node_Master mode defined in [R-DTI]. Values 0, 4-255 are reserved. The default value is RpdNodeSlave (2).

The RPD MUST support Node_Slave mode ("RpdNodeSlave" setting). The requirements for supporting Node_Master mode will be defined in a future version of this specification.

B.5.11.1.2 RpdPtpDefDsDomainNumber

RpdPtpDefDsDomainNumber is a TLV used to configure the identifier of the administrative domain in which RPD RDTI client operates. This attribute corresponds to defaultDS.domainNumber defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.2	1		R/W	An unsigned byte value representing the identifier of the administrative domain in which RPD RDTI client operates. G.8275.2 profile permits value range of 44-63. G.8275.2 profile defines a default value of 44.

B.5.11.1.3 RpdPtpDefDsPriority1

RpdPtpDefDsPriority1 is a TLV used to configure Priority1 attribute in the RPD RDTI client. This TLV is equivalent to defaultDS.priority1 attribute defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.3	1		R/W	An unsigned byte value used in selection of master clock. G.8275.2 profile does not use this attribute for slave operation. G.8275.2 profile defines a default value of 128.

B.5.11.1.4 RpdPtpDefDsPriority2

RpdPtpDefDsPriority2 is a TLV used to configure Priority2 attribute in the RPD RDTI client. This TLV is equivalent to defaultDS.priority2 attribute defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.4	1		R/W	An unsigned byte value used in selection of master clock. G.8275.2 profile defines a single value of this attribute (255) for slave operation. G.8275.2 profile defines a default value of 255 for slave operation.

B.5.11.1.5 RpdPtpDefDsLocalPriority

RpdPtpDefDsLocalPriority is a TLV used to configure LocalPriority attribute in RPD RDTI client. This TLV is equivalent to defaultDS.localPriority, a new data set member defined in G.8275.2.

TLV Type	Length	Units	Access	Value
97.5	1		R/W	An unsigned byte value assigned to the RPD RDTI client, to be used if needed when the data associated with the local clock, is compared with data on another potential GM received via an Announce message. G.8275.2 profile defines a range of values from 1 to 255 for slave operation. G.8275.2 profile defines a default value of 128 for slave operation.

B.5.11.1.6 RpdPtpProfileIdentifier

RpdPtpProfileIdentifier is a TLV used to configure the PTP profile for the RPD RDTI client.

TLV Type	Length	Units	Access	Value
97.6	6		R/W	A MAC Address uniquely identifying the configured PTP profile. For example G.8275.2 profile is designated by "00-19-A7-02-01-00". The default value is Null MAC address.

B.5.11.1.7 RpdPtpProfileVersion

RpdPtpProfileVersion is a TLV used to configure the PTP profile version to be used by the RPD RDTI client.

TLV Type	Length	Units	Access	Value
97.7	3		R/W	A 3 octet long hex string identifying the version of the configured PTP profile. The 3 byte string consists of two fields: A primaryVersion (Unsigned Short value) and a revisionNumber (Unsigned Byte value). For example G.8275.2 profile version is designated by "00-01-00". The default value is "00-00-00".

B.5.11.2 *RpdPtpPortConfig*

RpdPtpPortConfig is a complex TLV used to communicate attributes related to configuration of the PTP port in RDTI client in the RPD.

TLV Type	Length	Units	Access	Value
97.8	variable		R/W	A sequence of sub-TLVs representing configuration attributes of a PTP port in the RPD RDTI client.

B.5.11.2.1 RpdEnetPortIndex

RpdEnetPortIndex is a TLV identifying a CIN facing Ethernet port in the RPD. The sender MUST place the RpdEnetPortIndex TLV TLV as the first sub-TLV of the TLV 97.8.

TLV Type	Length	Units	Access	Value
97.8.1	2		N/A	An unsigned short value identifying Ethernet port in the RPD. There is no default value defined.

B.5.11.2.2 RpdPtpPortIndex

RpdPtpPortIndex is a TLV identifying a PTP port within the CIN facing Ethernet port in the RPD. The sender MUST place the RpdPtpPortIndex TLV as the second sub-TLV of the TLV 97.8.

TLV Type	Length	Units	Access	Value
97.8.2	2		N/A	An unsigned short value identifying PTP port on the Ethernet port in the RPD. There is no default value defined.

B.5.11.2.3 RpdAdminState

RpdPtpPortAdminState is a TLV used to configure the administrative status of the PTP port in the RPD.

TLV Type	Length	Units	Access	Value
97.8.3	1		R/W	An unsigned byte value indicating the administrative state of the PTP Port. The defined values are: 1 - "other" 2 - "up" 3 - "down" 4 - "testing" Values 0, 5-255 are reserved. The default value is "down" (2).

B.5.11.2.4 RpdPtpPortClockSource

RpdPtpPortClockSource is a TLV used to configure IP address of the primary PTP Master to which the PTP Slave needs to synchronize to. The RPD does not utilize master discovery protocol defined in [IEEE 1588]

TLV Type	Length	Units	Access	Value
97.8.4	4 or 16		R/W	The IP address of the PTP Master. The length signifies whether the value is an IPv4 or IPv6 address

B.5.11.2.5 RpdPtpPortAlternateClockSource

RpdPtpPortAlternateClockSource is a TLV used to configure IP address of the alternate PTP Master to which the PTP Slave in the RPD needs to synchronize to in case the connection to the primary clock source cannot be established.

TLV Type	Length	Units	Access	Value
97.8.5	4 or 16		R/W	The IP address of the alternate PTP Master. The length signifies whether the value is an IPv4 or IPv6 address

B.5.11.2.6 RpdPtpPortClockSelectAlternateSourceFirst

RpdPtpPortClockSelectAlternateSourceFirst is a TLV used to instruct the RDTI client to inverse the order of PTP Master source selection.

TLV Type	Length	Units	Access	Value
97.8.6	1		R/W	A Boolean value indicating whether the RDTI client needs to inverse the PTP Master selection. 0 - The RPD attempts to contact the primary PTP Master first. 1 - The RPD attempts to contact the alternate PTP Master first. Default value: 0.

B.5.11.2.7 RpdPtpPortTransportType

RpdPtpPortTransportType TLV configures PTP port's transport type.

TLV Type	Length	Units	Access	Value
97.8.7	1		R/W	An unsigned byte value defining PTP port transport. The defined values are: 1 - "other" 2 - "IPv4" 3 - "IPv6" Values 0, 4-255 are reserved. The default value is "Ipv4" (2).

B.5.11.2.8 RpdPtpPortTransportCos

RpdPtpPortTransportCos TLV configures PTP port's Class of Service (CoS) for usage in 802.1q VLAN tags in transmitted PTP packets.

TLV Type	Length	Units	Access	Value
97.8.8	1		R/W	An unsigned byte specifies the CoS value to be used in 802.1q tags. The range of permitted values is 0-7. The default value is 6 (Internetwork Control).

B.5.11.2.9 RpdPtpPortTransportDscp

RpdPtpPortTransportDscp TLV configures DSCP value for usage in IP headers of transmitted PTP packets.

TLV Type	Length	Units	Access	Value
97.8.9	1		R/W	An unsigned byte specifies the DSCP value to be used in IP headers of transmitted PTP packets. The range of permitted values is 0-63. The default value is 47 (Special Expedited Forwarding).

B.5.11.2.10 RpdPtpPortDsLocalPriority

RpdPtpDefDsLocalPriority is a TLV used to configure LocalPriority attribute in RPD PTP Port. This TLV is equivalent to portDS.localPriority, a new data member defined in G.8275.2.

TLV Type	Length	Units	Access	Value
97.8.10	1		R/W	An unsigned byte value assigned to the RPD PTP port, to be used as defined in G.8275.2. G.8275.2 profile defines a range of values from 1 to 255 for slave operation. G.8275.2 profile defines a default value of 128 for slave operation.

B.5.11.2.11 RpdPtpPortDsLogSyncInterval

RpdPtpPortDsLogSyncInterval TLV configures interval/frequency of Sync messages sent from the PTP port. This TLV is equivalent to attribute portDS.logSyncInterval defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.8.11	1		R/W	A signed byte value is logarithm to the base 2 of Sync interval measured in seconds. G.8275.2 defines range of values from 0 to -7 (1 sec to 1/128 sec) There is no default value defined.

B.5.11.2.12 RpdPtpPortDsLogAnnounceInterval

RpdPtpPortDsLogAnnounceInterval TLV configures the interval/frequency of Announce messages. This TLV is equivalent to attribute portDS.logAnnounceInterval defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.8.12	1		R/W	A signed byte value is logarithm to the base 2 of Announce interval measured in seconds. G.8275.2 defines range of values from 0 to -3 (1 sec to 1/8 sec). There is no default value defined.

B.5.11.2.13 RpdPtpPortDsLogDelayReqInterval

RpdPtpPortDsLogDelayReqInterval TLV configures the interval/frequency of Delay Request messages. This TLV is equivalent to attribute portDS.logMinDelayReqInterval defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.8.13	1		R/W	A signed byte value is logarithm to the base 2 of Delay Request interval measured in seconds. G.8275.2 defines range of values from 0 to -7 (1 sec to 1/128 sec) There is no default value defined.

B.5.11.2.14 RpdPtpPortDsAnnounceReceiptTimeout

RpdPtpPortDsAnnounceReceiptTimeout TLV configures the number of announce intervals before the session times out. This TLV is equivalent to attribute portDS.announceReceiptTimeout defined in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.8.14	1	seconds	R/W	The unsigned byte value has a range of 3-255 seconds. There is no default value defined.

B.5.11.2.15 RpdPtpPortUnicastContractDuration

RpdPtpPortUnicastContractDuration TLV configures the interval for which the PTP port requests unicast service. There is equivalent attribute defined in port dataset in [IEEE 1588].

TLV Type	Length	Units	Access	Value
97.8.15	2	seconds	R/W	The unsigned short value has a range of 60 to 1000 seconds. The default value is 300 seconds.

B.5.11.2.16 RpdPtpPortClockSrcGw

RpdPtpPortClockSrcGw attribute allows the CCAP Core to configure a gateway address for PTP traffic sent from the specified PTP port to the primary PTP Master. When the RpdPtpPortClockSrcGw is Null (0.0.0.0) the RPD uses the same gateway address as for all other traffic originating from the corresponding Ethernet port.

TLV Type	Length	Units	Access	Value
97.8.16	4 | 16	N/A	R/W	An IP address of a gateway through which the RPD can reach the primary PTP Master from the selected PTP port. The default value is Null IP address (0.0.0.0).

B.5.11.2.17 RpdPtpPortClockAltSrcGw

RpdPtpPortClockSrcAltGw attribute allows the CCAP Core to configure a gateway address for PTP traffic sent from the specified PTP port to the alternate PTP Master. When the RpdPtpPortClockSrcAltGw is Null (0.0.0.0) the RPD uses the same gateway address as for all other traffic originating from the corresponding Ethernet port.

TLV Type	Length	Units	Access	Value
97.8.17	4 | 16	N/A	R/W	An IP address of a gateway through which the RPD can reach the alternate PTP Master from the selected PTP port. The default value is Null IP address (0.0.0.0).

B.6 RPD Operational Monitoring[96]

Table 22 provides the summary of GCP TLV encodings for RPD operational monitoring

Table 22 - Summary of GCP TLV Encodings for RPD Operational Monitoring

Attribute/Class Name	Object Type	TLV Type	TLV Value Field Length	Mandatory	Constraints	Comments
RpdInfo	ComplexTLV	100	-		R	See CM SP R-OSSI for all GCP TLV 100.x assignments
DOCS-IF31	Complex TLV	101			R	Reserved for DOCS-IF31 objects
DOCS-IF3	Complex TLV	102			R	Reserved for DOCS-IF3 objects
DOCS-IF	Complex TLV	103			R	Reserved for DOCS-IF objects
HOST-RESOURCES	Complex TLV	104			R	

[96] Section and table added per R-PHY-N-16.1476-1 on 4/19/16 by JB. Table replaced and Sections B.6.1 through B.6.25.12 deleted per R-PHY-N-16.1586-2 on 9/8/16 by JB.

Attribute/Class Name	Object Type	TLV Type	TLV Value Field Length	Mandatory	Constraints	Comments
hrMemorySize	KBytes	104.1	1..2147483647		R	
hrProcessorLoad	Integer32	104.2	1..2147483647		R	
hrStorageAllocationFailures	Counter32	104.3	4		R	
hrStorageAllocationUnits	Integer32	104.4	1..2147483647		R	
hrStorageIndex	Integer32	104.5	1..2147483647		R	
hrStorageSize	Integer32	104.6	1..2147483647		R	
hrStorageType	AutonomousType	104.7			R	
hrStorageUsed	Integer32	104.8	1..2147483647		R	
hrSWRunPerfCPU	Integer32	104.9	1..2147483647		R	
hrSWRunPerfMem	KBytes	104.10	1..2147483647		R	
hrSWRunStatus	Integer	104.11			R	
hrSWRunType	Integer	104.12			R	

B.6.1 Output Buffer Occupancy History and Buffer Depth Monitoring TLVs[97]

[DEPI] specification outlines two functions which allow the CCAP Core to monitor the status of the output queues in the RPD. These functions are Downstream Output Buffer History and Downstream Buffer Depth Monitoring Alerts. For this purpose, the RPC defines two TLVs which are explained further in this section.

B.6.1.1 *Output Buffer Occupancy History*

OutputBufferOccupancyHistory is a complex TLV which is used to configure buffer depth in RPD and to retrieve buffer occupancy history information from the RPD.

TLV Type	Length	Units	Access	Value
83	variable		N/A	A sequence of sub-TLVs used to configure buffer depth and retrieve buffer occupancy monitoring information from the RPD.

B.6.1.1.1 MaximumBufferDepth

MaximumBufferDepth TLV allows the CCAP Core to read the maximum output buffer depth supported by the RPD for a selected channel.

TLV Type	Length	Units	Access	Value
83.1	4	bytes	R/O	An unsigned integer value specifying the maximum buffer depth supported by the RPD for the selected downstream channel.

B.6.1.1.2 CurrentBufferDepth

CurrentBufferDepth TLV allows the CCAP Core to read the current depth of output buffer as well as to configure the actual output buffer depth for the selected channel if the RPD supports such option. The RPD communicates its support for configurable buffer depth for DOCSIS channels through a defined capability (BufferDepthConfigurationSupport).

If the RPD does not support configurable output buffer depth for a particular channel type, the CCAP Core MUST NOT write to this attribute.

[97] New section added per R-PHY-N-16.1576-1 on 9/19/16 by JB.

TLV Type	Length	Units	Access	Value
83.2	4	Bytes	R/W	An unsigned integer value specifying the buffer depth for the selected downstream channel in bytes.

B.6.1.1.3 EnableMonitor

EnableMonitor TLV is used by CCAP Core to enable and disable output buffer depth monitoring for a selected channel in the RPD.

TLV Type	Length	Units	Access	Value
83.3	1	-	R/W	A Boolean value specifying whether monitoring is enabled for the specified channel. Defined values are: 0 - Monitoring is disabled. 1 - Monitoring is enabled.

B.6.1.1.4 NormalizationFactor

NormalizationFactor TLV is used by CCAP Core to provide a Normalization Factor (NF) to the RPD for the selected channel. The RPD converts buffer depth measurements from values expressed in bytes to normalized 8-bit samples by dividing measurements expressed in bytes by the Normalization Factor. The Normalization Factor detailed definition is provided in [DEPI].

TLV Type	Length	Units	Access	Value
83.4	4		R/W	An unsigned integer value specifying the normalization factor.

B.6.1.1.5 FirstSampleTimestamp

When the CCAP Core reads the buffer occupancy history, the RPD reports the time when the first sample was collected via FirstSampleTimestamp TLV.

TLV Type	Length	Units	Access	Value
83.5	4		R/O	32-bit DOCSIS timestamp indicating the time when the first sample in the buffer occupancy history was collected.

B.6.1.1.6 SampledBufferOccupancy

SampledBufferOccupancy TLV is used by the CCAP Core to read the buffer occupancy history from the RPD.

TLV Type	Length	Units	Access	Value
83.6	1000		R/O	A sequence of 1000 8-bit unsigned byte values starting with the oldest sample and ending with a newest sample. Each value provides a sample of measured output queue depth in normalized units.

B.6.1.2 *Downstream Buffer Depth Monitoring Alerts*

OutputBufferThresholdAlert is a complex TLV which is used to configure RPD to monitor output queue depth and to send alerts to the CCAP Core.

TLV Type	Length	Units	Access	Value
84	variable		N/A	A sequence of sub-TLVs used to configure and retrieve output queue depth monitoring alerts.

B.6.1.2.1 BufferDepthMonAlertEnable

BufferDepthMonAlertEnable TLV is used by the CCAP Core to enable or disable buffer depth monitoring alert in the RPD.

TLV Type	Length	Units	Access	Value
84.1	1		R/W	A Boolean value indicating the administrative status of the buffer depth monitoring alert function in the RPD for the selected channel. The following values have been defined: 0 - disabled. 1 - enabled. All other values are reserved.

B.6.1.2.2 BufferDepthMonAlertStatus

The CCAP Core uses BufferDepthMonAlertStatus TLV to read the status of the buffer depth monitoring alert in the RPD.

TLV Type	Length	Units	Access	Value
84.2	1		R/W	An unsigned byte value indicating the operational status of the buffer depth monitoring alert function in the RPD for the selected channel. The following values have been defined: 1 - “other”, 2 - “idle”, 3 - “running”, 4 - “threshold exceeded”. Values 0, 5-255 are reserved.

B.6.1.2.3 AlertThreshold

The CCAP Core uses BufferDepthMonAlertStatus TLV to configure the threshold value for the output queue depth monitor in the RPD.

TLV Type	Length	Units	Access	Value
84.3	1		R/W	An unsigned byte indicating the programmed threshold value in normalized units (1-255).

B.6.1.2.4 SmoothingFactorN

SmoothingFactorN TLV is used by the CCAP Coreto configure the exponential value ‘N’ for the purpose of computing the smoothing factor α utilized in computing the exponential moving average of the output queue depth.

TLV Type	Length	Units	Access	Value
84.4	1		R/W	An unsigned byte value N.

B.6.1.2.5 LastAlertTimestamp

LastAlertTimestamp TLV is used by the CCAP Core to read the time when the threshold of the output queue depth (calculated as EMA) has been exceeded and when the last alert was sent.

TLV Type	Length	Units	Access	Value
84.5	4		R/O	An unsigned integer value indicating a 32-bit DOCSIS timestamp when the threshold of the output queue depth (calculated as EMA) has been exceeded.

B.6.1.3 *Event Notification TLVs*

B.6.1.3.1 EventNotification

EventNotification is a complex TLV used by the CCAP Core to read event reports from the RPD and by the RPD to send event reports to the CCAP Core.

TLV Type	Length	Units	Access	Value
85	Variable		N/A	A set of sub-TLVs for attributes of RPD event reports.

B.6.1.3.2 RpdEvLogIndex

The Principal CCAP Core uses RpdEvLogIndex attribute to select the index of the first entry when reading the RPD's Local Event Log. This TLV is not used when the RPD sends event reports to the CCAP Core via Notify Message or when reading the Pending Event Report Queue.

TLV Type	Length	Units	Access	Value
85.1	4		N/A	An unsigned integer value specifying an index to the RPD Local Event Log. The range of supported values is from 0 to RPD Local Event Log Size minus 1, RPD Local Event Log Size is reported by the RPD in RpdLocalEventLogSize capability.

B.6.1.3.3 PendingOrLocalLog

This attribute allows the CCAP Core to select between Pending Event Report Queue and RPD Local Event Log when reading event reports.

When issuing a read request, the CCAP Core MUST include PendingOrLocalLog TLV to select between the Pending Event Report Queue and the RPD Local Event Log as the target of the read request. The RPD MUST NOT use this TLV when sending event reports via Notify message.

TLV Type	Length	Units	Access	Value
85.2	1		N/A	An unsigned byte value. The permitted values are: 0 - the target of the read request is the Pending Event Report Queue. 1 - the target of the read request is the Local Event Log.

B.6.1.3.4 EvFirstTime

EvFirstTime attribute indicates the time when the event has occurred first time.

TLV Type	Length	Units	Access	Value
85.3	8 | 11		R	An octet string. The value field format is defined in [RFC 4639]. The reported value is based on RPD clock.

B.6.1.3.5 EvLastTime

EvLastTime attribute indicates the last time when the event has occured.

TLV Type	Length	Units	Access	Value
85.4	8 |11		R	An octet string. The value field format is defined in [RFC 4639]. The reported value is based on RPD clock.

B.6.1.3.6 EvCounts

EvCounts attribute indicates the number of times an event has occured. When RPD sends event reports via Notify message or places event reports in the Pending event report queue, the RPD MUST indicate the number of new occurences of the event since the last time an event report was issued for this event.

TLV Type	Length	Units	Access	Value
85.5	4		R	An unsigned integer value indicating the number of event occurences.

B.6.1.3.7 EvLevel

EvLevel attribute indicates the event priority level.

TLV Type	Length	Units	Access	Value
85.6	4		R	An unsigned byte value specifying the priority level for event. The valid values are defined in RFC4639 and listed here for easier reference: 1 - emergency 2 - alert 3 - critical 4 - error 5 - warning 6 - notice 7 - information 8 - debug

B.6.1.3.8 EvtId

EvId attribute indicates the event identifier which uniquely determes the type of event.

TLV Type	Length	Units	Access	Value
85.7	4		R	An unsigned integer value indicating the event identifier. RPD event identifiers are defined in [R-OSSI]. There is no defualt value defined.

B.6.1.3.9 EvString

EvString attribute is used for a human-readable description of the event, including all relevant context.

TLV Type	Length	Units	Access	Value
85.8	1-255		R	A string with length of 1-255 octets. The detailed requirements for the formatting of EvString are defined in [R-OSSI].

Annex C MPEG Stream Analysis (Normative)

The RPD MAY support MPEG Stream Analysis as described in this annex.

In order to validate that the MPEG stream served from the CCAP Core does not have issues that will cause video outages or other service impairments, the RPD will optionally be capable of performing tests on an MPEG stream to verify its integrity. These checks are designed to detect video disruption and outages by detecting Packet Identifier (PID) discontinuities and PID bitrate (or PID count) thresholds. The RPD monitors PIDs within both multi-program and single-program transport streams (MPTS and SPTS) used to carry various MPEG system and control information, video payloads, and audio payloads. PIDs monitored include the Program Association Table (PAT), the Program Map Table (PMT), Program and System Information Protocol (PSIP), 0x1FFC carousel, and PIDs within a PMT program such as video, audio, SCTE-35/Digital Program Insertion (DPI), and Enhanced TV Binary Interchange Format (EBIF).

If the RPD supports MPEG Stream Analysis, it MUST monitor MPEG synchronization by detecting transport stream synchronization loss. A device synchronizes on a transport stream via the reception of correct sync bytes, which are the 8 bits that precede the header of an MPEG packet (always 0x47). When the decoder first detects the sync byte, it looks again for the next sync byte after 188 or 204 bytes in the stream. After finding three sync bytes in a row in this pattern, synchronization has been established and packet boundaries are then known. However, if packets arrive with incorrect sync bytes, synchronization loss occurs and the decoder again establishes MPEG synchronization. The RPD will consider synchronization lost when two or more consecutive incorrect sync bytes are received.

Once the RPD has achieved MPEG synchronization, the following evaluations can be performed:

- If the RPD supports MPEG Stream Analysis, it MUST be capable of reading the transport stream ID (TSID) from the PAT. This value can be reported in enterprise MIBs. Note that a TSID is not available in DOCSIS streams.
- If the RPD supports MPEG Stream Analysis, it MUST detect program PID discontinuity resulting in media loss.
- If the RPD supports MPEG Stream Analysis, it MUST be capable of detecting the existence of the following program PIDs in the transport stream:
    - PAT
    - PMT
    - Video
    - Audio
    - SCTE-35/DPI
    - EBIF
- When loss of one of these PIDs is detected, it is expected that the RPD, using an SNMP trap or Syslog event, will provide notification to the operator.
- If the RPD supports MPEG Stream Analysis, it MUST be capable of detecting the existence of the mini-carousel PID (0x1FEE) and reading the service group ID (SGID) from the PID.

In addition, the bit rates of certain PIDs can provide insight into the health of a given stream. For example, a too-low bit rate could mean failure of the component providing the PID stream; a too-high bit rate could indicate an error condition on that device. Either of these occurrences can cause service disruption. The rates of these can be monitored via counters and a rate calculated on a time scale of minutes or several minutes to determine the health of the stream. If the RPD supports MPEG Stream Analysis, it MUST monitor the PID bit rate of the following PIDs:

- DOCSIS PID 0x1FFE
- ATSC A65 PSIP base PID 0x1FFB
- In-band DTA PIDs, including SI PID, 0x1FFC, and 0x1FF0

When an abnormal bit rate is detected, it is expected that the RPD, using an SNMP trap or Syslog event, will provide notification to the operator.

Annex D Certificate Hierarchy and Profiles (Normative)[98]

This section describes the certificate format and extensions used by CableLabs certification authorities (CA) and summarizes the fields of [X.509] version 3 certificates used for this specification. The CableLabs certificate PKI hierarchy is shown below:

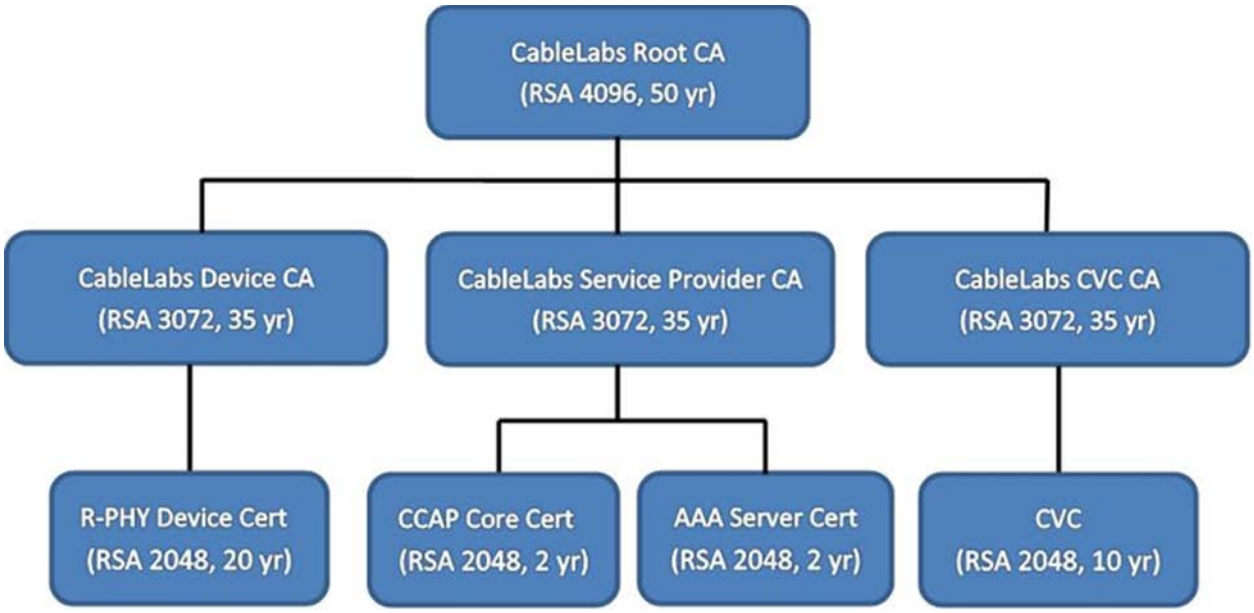

Figure 42 - Certificate Hierarchy

All certificates and CRLs described in this specification are signed with the RSA signature algorithm, using SHA-256 as the hash function. The RSA signature algorithm is described in PKCS #1 [RSA 1]; SHA-256 is described in [FIPS 180-4].

NOTE: intermediate CA certificate Subject DN attributes may change due to maintanence/management of the PKI which would also cause the Issuer DN attributes to change of end entity certificates.

D.1 CableLabs Root CA Certificate

The contents of the CableLabs Root CA Certificate is shown in Table 23.

Table 23 - CableLabs Root CA Certificate

Attribute Name	Settings
Version	v3
Serial number	Unique Positive Integer assigned by the CA
Issuer DN	c=US o=CableLabs ou=Root CA01 cn=CableLabs Root Certification Authority
Subject DN	c=US o=CableLabs ou=Root CA01 cn=CableLabs Root Certification Authority
Validity Period	50 yrs
Public Key Algorithm	Sha256WithRSAEncryption (1 2 840 113549 1 1 11)

[98] Revised per R-PHY-N-16.1514-1 on 8/10/16 by JB.

Attribute Name		Settings		
Keysize		4096 bits		
Parameters		NULL		
Standard Extensions	OID	Include	Criticality	Value
keyUsage	{id-ce 15}	X	TRUE	
keyCertSign				Set
cRLSign				Set
basicConstraints	{id-ce 19}	X	TRUE	
cA				Set
subjectKeyIdentifier	{id-ce 14}	X	FALSE	
keyIdentifier				Calculated per Method 1
subjectAltName	{id-ce 17}	O	FALSE	
directoryName				Set by the issuing CA

D.2 CableLabs Device CA Certificate

The contents of the CableLabs Device CA Certificate are shown in Table 24.

Table 24 - CableLabs Device CA Certificate

Attribute Name		Settings		
Version		v3		
Serial number		Unique Positive Integer assigned by the CA		
Issuer DN		c=US o=CableLabs ou=Root CA01 cn=CableLabs Root Certification Authority		
Subject DN		c=US o=CableLabs ou=Device CA01 cn=CableLabs Device Certification Authority		
Validity Period		35 yrs		
Public Key Algorithm		Sha256WithRSAEncryption (1 2 840 113549 1 1 11)		
Keysize		3072-bits		
Parameters		NULL		
Standard Extensions	OID	Include	Criticality	Value
keyUsage	{id-ce 15}	X	TRUE	
keyCertSign				Set
cRLSign				Set
basicConstraints	{id-ce 19}	X	TRUE	
cA				Set
pathLenConstraint				0
subjectKeyIdentifier	{id-ce 14}	X	FALSE	
keyIdentifier				Calculated per Method 1
authorityKeyIdentifier	{id-ce 35}	X	FALSE	
keyIdentifier				Calculated per Method 1
subjectAltName	{id-ce 17}	O	FALSE	
directoryName				Set by the issuing CA for online CAs

D.3 RPD Certificate

The contents of the RPD Certificate are shown in Table 25.

Table 25 - RPD Certificate[99]

Attribute Name		Settings		
Version		v3		
Serial number		Unique Positive Integer assigned by the CA		
Issuer DN		c=US o=CableLabs ou=Device CA01 cn=CableLabs Device Certification Authority		
Subject DN		c=<Country of Manufacturer> o=<Company Name> ou=<Manufacturing Location> cn=<MAC Address>		
Validity Period		20 yrs		
Public Key Algorithm		Sha256WithRSAEncryption (1 2 840 113549 1 1 11)		
Keysize		2048 bits		
Parameters		NULL		
Standard Extensions	OID	Include	Criticality	Value
keyUsage	{id-ce 15}	X	TRUE	
digitalSignature				Set
keyEncipherment				Set
authorityKeyIdentifier	{id-ce 35}	X	FALSE	
keyIdentifier				Calculated per Method 1

Values in angle brackets (<>) indicate that appropriate text as indicated below is present:

<Country of Manufacturer>: two-letter country code;

<Company Name>: name that identifies the company;

<MAC Address>: MAC address of the RPD.

The MAC Address is expressed as six pairs of hexadecimal digits separated by single colons (:), e.g., `00:60:21:A5:0A:23`. Hexadecimal digits greater than 9 are expressed as uppercase letters.

D.4 CableLabs Service Provider CA Certificate

The contents of the CableLabs Service Provider Certificate are shown in Table 26.

Table 26 - CableLabs Service Provider CA Certificate[100]

Attribute Name	Settings
Version	v3
Serial number	Unique Positive Integer assigned by the CA
Issuer DN	c=US o=CableLabs ou=Root CA01 cn=CableLabs Root Certification Authority
Subject DN	c=US

[99] Revised per R-PHY-N-16.1514-1 on 8/10/16 by JB.
[100] Revised per R-PHY-N-16.1514-1 on 8/10/16 by JB.

Attribute Name		Settings		
		o=CableLabs ou=Service Provider CA01 cn=CableLabs Service Provider Certification Authority		
Validity Period		Up to 35 yrs		
Public Key Algorithm		Sha256WithRSAEncryption (1 2 840 113549 1 1 11)		
Keysize		3072 bits		
Parameters		NULL		
Standard Extensions	OID	Include	Criticality	Value
keyUsage	{id-ce 15}	X	TRUE	
keyCertSign				Set
cRLSign				Set
basicConstraints	{id-ce 19}	X	TRUE	
cA				Set
pathLenConstraint				0
subjectKeyIdentifier	{id-ce 14}	X	FALSE	
keyIdentifier				Calculated per Method 1
authorityKeyIdentifier	{id-ce 35}	X	FALSE	
keyIdentifier				Calculated per Method 1
subjectAltName	{id-ce 17}	O	FALSE	
directoryName				Set by the issuing CA for online CAs

D.5 AAA Server Certificate and CCAP Core Certificate[101]

The contents of the AAA Server Certificate and CCAP Core Certificate are shown in Table 27.

Table 27 - AAA Server Certificate and CCAP Core Certificate

Attribute Name		Settings		
Version		v3		
Serial number		Unique Positive Integer assigned by the CA		
Issuer DN		c=US o=CableLabs ou=Service Provider CA01 cn=CableLabs Service Provider Certification Authority		
Subject DN		c=<Country> o=<Company Name> cn=<server FQDN>		
Validity Period		2 yrs		
Public Key Algorithm		Sha256WithRSAEncryption (1 2 840 113549 1 1 11)		
Keysize		2048 bits		
Parameters		NULL		
Standard Extensions	OID	Include	Criticality	Value
keyUsage	{id-ce 15}	X	TRUE	
digitalSignature				Set
keyEncipherment				Set

[101] [101] Revised per R-PHY-N-16.1514-1 on 8/10/16 by JB.

Attribute Name		Settings		
authorityKeyIdentifier	{id-ce 35}	X	FALSE	
keyIdentifier				Calculated per Method 1
subjectAltName	{id-ce 17}	X	FALSE	
dNSName				<server FQDN>

Annex E Receive Power Level Management (Normative)[102]

This annex describes the physical layer RF power level specifications required for the location and operation of a DOCSIS upstream demodulator in an optical node in a cable television plant, serving as additions and modifications to the DOCSIS PHY3.1 specifications. The DOCSIS upstream demodulator is part of a Remote PHY Device (RPD) module contained within an optical node, instead of being located at a headend or hub site.

E.1 Problem Definition, Scope and Purpose

E.1.1 Problem Definition

In today's DOCSIS upstream the signal path is approximately as follows:

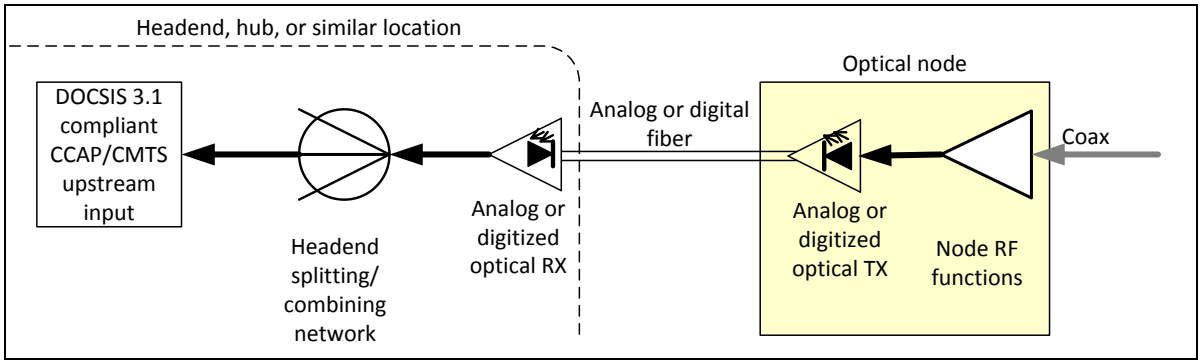

Figure 43 - Traditional Upstream RF Signal Path

In this case, the DOCSIS PHY requirements are dictated not only by the desired performance on the coax but also by the need to pass through a headend splitting network after reception of the optical signal at the headend. The splitting operations and the analog optical link (or the A/D and D/A converters of a digitized optical link) introduce various distortions and degradations which ultimately reduce the received signal quality.

A major objective of placing the PHY in the node is to improve the received signal quality by removing the analog (or digitized) optics. When the CMTS/CCAP input is located in the node, the signal path looks more like the diagram below:

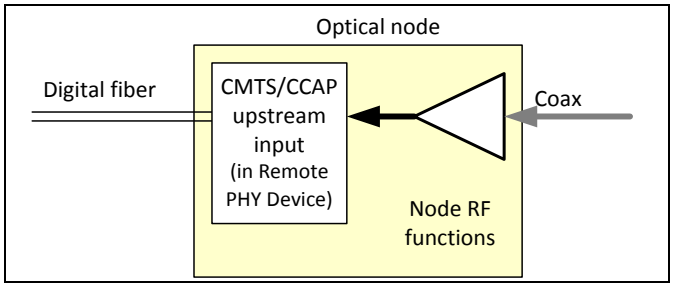

Figure 44 - R-PHY Upstream RF Signal Path

The shorter and cleaner signal path results in improved received signal quality, which helps to enable the use of the higher upstream modulation orders provided by DOCSIS 3.1 technology.

The main reason the CMTS receiver input power requirements of DOCSIS PHY3.1 cannot be mandated directly when the RPD is located in a node is because DOCSIS PHY3.1 was designed to accommodate the expected power level after the headend splitting network, as mentioned above. The power levels and range of overall power

[102] Revised per R-PHY-N-16.1616-1 on 12/19/16 by JB.

adjustability (not per-channel adjustability) needed at the headend are not necessarily the same as those needed at the output of the upstream RF functions of a fiber node.

This annex documents the variances from the DOCSIS PHY3.1 which are allowed/required when the CMTS/CCAP input is located in an RPD within a fiber node (this annex does not apply to an RPD in a headend or hub location).

For this effort, the RPD is modeled as a module within a fiber node. Node functionality not currently in scope for the R-PHY specifications is considered to be outside of the RPD module.

The objective of this annex is to maintain DOCSIS PHY3.1 performance levels at the RPD module input while allowing for variances that better match the node RF output to the RPD module input.

Figure 45 shows the model partitioning in more detail. This is intended as an example to illustrate the demarcation of the RPD module.

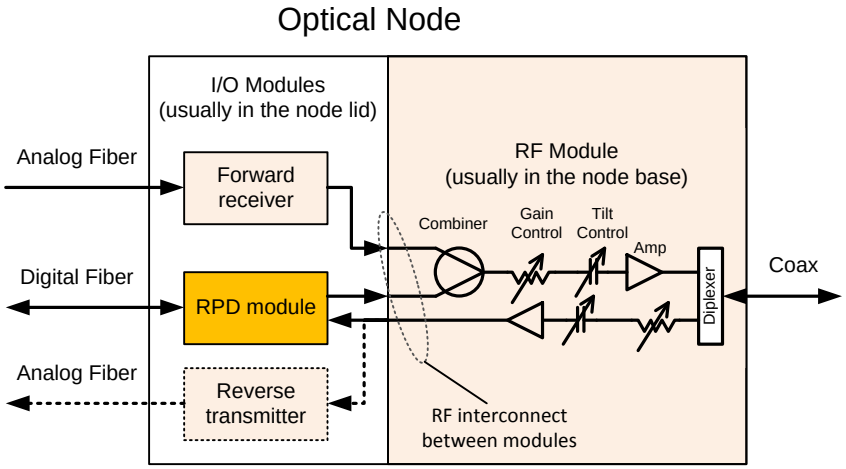

Figure 45 - R-PHY RF Interface Definition

From the node perspective, the RPD module looks like a forward optical receiver plus reverse optical transmitter, with an RF interface between the module and the rest of the node. This receiver/transmitter could be in addition to receivers/transmitters already present for other purposes (e.g., analog video, telemetry) as shown in the diagram, or it could be the only receiver/transmitter in the node. Various physical form factors and combinations of functionality outside of the RPD module can be supported without impacting the requirements of the RPD module itself.

It is recognized that a physical implementation of an RPD and fiber node may not exactly conform to the model described. In this case, for purposes of compliance testing, the vendor would be responsible for providing a test point and system configuration at which the available signal meets RPHY/DRFI Annex requirements.

E.1.2 Scope

This annex defines modifications to DOCSIS PHY3.1 specification which apply when the upstream demodulator, or PHY Device, is located in an optical node in a cable television plant.

DOCSIS PHY3.1 specification defines specifications for a CMTS which is presumed to be located at a headend or hub site. However, the DCA-MHAv2 family of specifications describes an alternate system architecture in which the upstream demodulator, or PHY Device, may instead be located within an optical node as part of an RPD. Most, but not all, of the requirements for a PHY Device in a DCA-MHAv2 architecture are the same as those which apply to a CMTS or EQAM.

E.1.3 Purpose

The purpose of this annex is to define the RF characteristics required in the upstream receiver of a PHY Device located within an optical node, with sufficient specificity to enable vendors to build devices meeting the needs of

cable operators around the world. This annex can be used by CableLabs to develop a certification/qualification program for such devices.

E.2 RPD Receive Power Level

In current HFC deployments, upstream power levels at the node input are normally significantly higher than the common headend CMTS operating levels of 0 dBmV per channel.

Typically employed node input upstream signal level ranges from as low as 8 dBmV per ATDMA channel to as high as 24 dBmV per ATDMA channel. At the input of a return transmitter, the upstream channel power level will typically be 0 to 10 dB higher than at the node input port.

The RPD MUST be settable according to Table 28 for intended received power normalized to 6.4 MHz of bandwidth. Table This requirement replaces the equivalent one from [PHYv3.1] Section 7.4.14.1, by referring to Table 28 instead of Table 7-12 of [PHYv3.1]. To clarify, an RPD that implements a range of settable (chosen) set points (per US RF port) across a portion of the range in Table 28 complies with this requirement, and so is an RPD that implements just a single fixed set point in this range.

The RPD upstream demodulator MUST operate within its defined performance specifications at any set point it supports, with received bursts within the ranges defined in Table 28 of the set power. This requirement replaces the equivalent one from [PHYv3.1] Section 7.4.14.1, by referring to Table 28 instead of Table 7-12 of [PHYv3.1].

Table 28- Upstream Channel Demodulator Input Power Characteristics

Modulation	Minimum Set Point (dBmV/6.4 MHz)	Maximum Set Point (dBmV/6.4 MHz)	Range
QPSK	-4 dBmV	25 dBmV	-9 / +3
8-QAM	-4 dBmV	25 dBmV	-9 / +3
16-QAM	-4 dBmV	25 dBmV	-9 / +3
32-QAM	-4 dBmV	25 dBmV	-9 / +3
64-QAM	-4 dBmV	25 dBmV	-9 / +3
128-QAM	0 dBmV	25 dBmV	-9 / +3
256-QAM	0 dBmV	25 dBmV	-9 / +3
512-QAM	0 dBmV	25 dBmV	-3 / +3
1024-QAM	0 dBmV	25 dBmV	-3 / +3
2048-QAM	7 dBmV	25 dBmV	-3 / +3
4096-QAM	10 dBmV	25 dBmV	-3 / +3

The RPD MUST meet the error ratio performance requirement of Section 7.4.14.2 of [PHYv3.1] at any set point it supports, (per US RF port), taken from Table 28 instead of Table 7-12 of [PHYv3.1], and with the same power spectral density for every D3.1 channel and the equivalent (same power spectral density) for every D3.0 channel.

E.3 Maximum Receive Composite Power Level

Due to the higher power levels, the node environment often does not comply with the conditions specified by [PHYv3.0] Section 6.2.23 Upstream Demodulator Input Power Characteristics:

"The instantaneous input signal level, including ingress and noise to the upstream demodulator, MUST NOT exceed 29 dBmV in the 5-85 MHz frequency range of operation."

Similarly, the node environment often does not comply with the slightly increased level specified in [PHYv3.1] Section 7.4.14.1:

“The CMTS Upstream Demodulator MUST operate with an average input signal level, including ingress and noise to the upstream demodulator, up to 31 dBmV. “

Due to the higher power level, especially with 204 MHz high split, the average composite signal power at the RPD input port can be as high as 35 dBmV, and the instantaneous composite power can be as high as 50 dBmV.

The RPD MUST operate with an average upstream input total composite power, including ingress and noise, up to 6 dB higher than the calculated total composite power based upon the chosen set point, as measured at interface C. This requirement replaces the [PHYv3.1] Section 7.4.14.1 absolute requirement of the CMTS to operate with up to 31 dBmV average power.

Annex F DOCSIS 3.1 OFDM Modifications for Remote PHY (Normative)[103]

This Annex describes the physical layer RF specifications required for the location and operation of a DOCSIS downstream OFDM modulator in an optical node in a cable television plant, serving as additions and modifications to the DOCSIS PHY3.1 specifications. The DOCSIS downstream modulator is part of a Remote PHY Device (RPD) module contained within an optical node, instead of being located at a headend or hub site.

F.1 Problem Definition, Scope and Purpose

F.1.1 Problem Definition

In today's DOCSIS downstream, the signal path is approximately as shown in Figure 46:

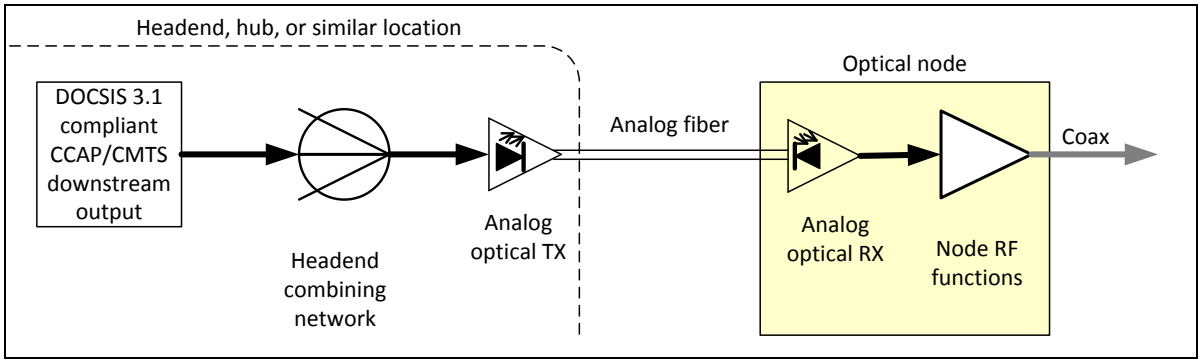

Figure 46 - Traditional Downstream RF Signal Path

In this case, the DOCSIS PHY requirements are dictated not only by the desired performance on the coax, and associated amplifiers and splitters, but also by the analog optical fiber link. There is also coax within the headend, and possibly combining networks. The splitting and/or combining operations and the analog optical link introduce various distortions and degradations which ultimately reduce the received signal quality.

A major objective of placing the PHY in the node is to improve the received signal quality by removing the analog optics. When the CMTS/CCAP output is located in the node, the signal path looks more like as illustrated in Figure 47:

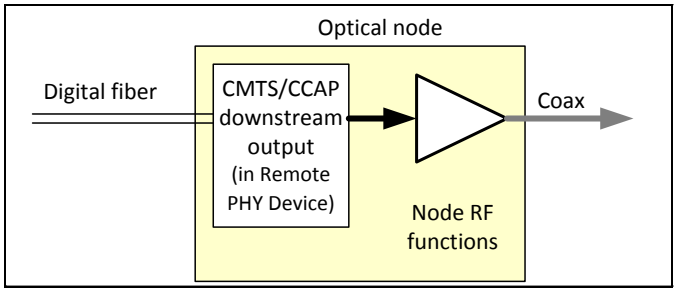

Figure 47 - R-PHY Downstream RF Signal Path

[103] Annex added per R-PHY-N-16.1663-1 on 12/20/16 by JB.

The shorter and cleaner signal path results in improved received signal quality, which helps to enable the use of the higher downstream modulation orders provided by DOCSIS 3.1 technology.

The main reason the CMTS downstream RF output requirements of DOCSIS PHYv3.1 cannot be mandated directly when the RPD is located in a node is because DOCSIS PHY3.1 was designed to accommodate the expected power level at input to the optical transmitter, after transmission through coax and possibly a combining network within the headend, as mentioned above. The power levels and range of overall power adjustability (not per-channel adjustability) needed at the headend are not necessarily the same as those needed at the output of the downstream RF functions within the fiber node.

This Annex documents the variances from the DOCSIS PHY3.1 which are allowed/required when the CMTS/CCAP output is located in an RPD within a fiber node (this Annex does not apply to an RPD in a headend or hub location).

For this effort, the RPD is modeled as a module within a fiber node. Node functionality not currently in scope for the R-PHY specifications is considered to be outside of the RPD module.

The objective of the Annex is to maintain DOCSIS PHY3.1 performance levels at the RPD module output while allowing for variances that better match the node RF output to the RPD module input.

Figure 48 shows the model partitioning in more detail, and indicates the Interface C within the node where the requirements of this Annex apply. This is intended as an example to illustrate the demarcation of the RPD module.

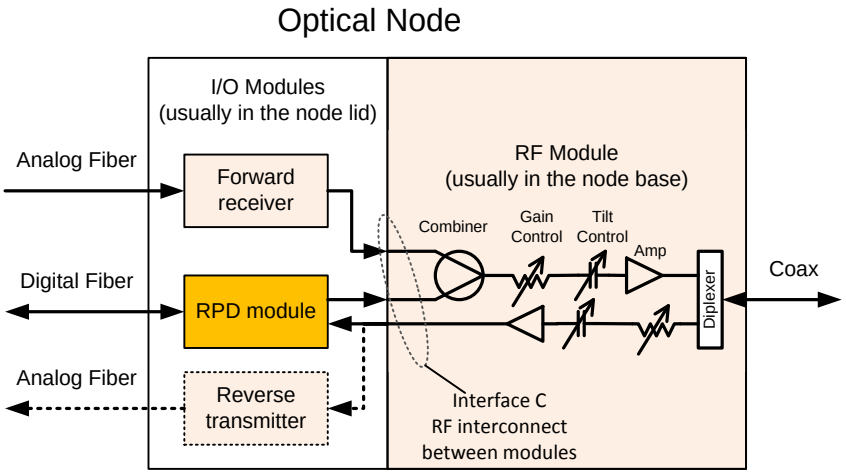

Figure 48 - R-PHY RF Interface Definition

From the node perspective, the RPD module looks like a forward optical receiver plus reverse optical transmitter, with an RF interface between the module and the rest of the node. This receiver/transmitter could be in addition to receivers/transmitters already present for other purposes (e.g., analog video, telemetry) as shown in the figure, or it could be the only receiver/transmitter in the node. Various physical form factors and combinations of functionality outside of the RPD module can be supported without impacting the requirements of the RPD module itself.

It is recognized that a physical implementation of an RPD and fiber node may not exactly conform to the model described. In this case, for purposes of compliance testing the vendor would be responsible for providing a test point and system configuration at which the available signal meets RPHY/DRFI Annex requirements.

F.1.2 Scope

This Annex defines modifications to DOCSIS PHY3.1 specification which apply when the downstream modulator, or PHY Device, is located in an optical node in a cable television plant.

DOCSIS PHY3.1 specification defines specifications for a CMTS which is presumed to be located at a headend or hub site. However, the DCA-MHAv2 family of specifications describes an alternate system architecture in which the downstream modulator, or PHY Device, may instead be located within an optical node as part of an RPD (Remote PHY Device). Most, but not all, of the requirements for a PHY Device in a DCA-MHAv2 architecture are the same as those which apply to a CMTS or EQAM.

F.1.3 Purpose

The purpose of this Annex is to define the RF characteristics required in the OFDM downstream modulator of a PHY Device located within an optical node, with sufficient specificity to enable vendors to build devices meeting the needs of cable operators around the world. This Annex can be used by CableLabs to develop a certification/qualification program for such devices.

F.2 Fidelity Requirements

For the purposes of this specification, the number of occupied CTA channels of an OFDM channel is the occupied bandwidth of the OFDM channel divided by 6 MHz.

RPDs capable of generating N-channels of legacy DOCSIS plus NOFDM-channels of OFDM per RF port, for purposes of the DRFI output electrical requirements, the device is said to be capable of generating Neq-channels per RF port, where Neq = N + 32*NOFDM "equivalent legacy DOCSIS channels."

An Neq-channel per RF port RPD MUST comply with all requirements operating with all Neq channels on the RF port, and MUST comply with all requirements for an Neq'-channel per RF port device operating with Neq' active channels on the RF port for all values of Neq' less than Neq.

For an OFDM channel there is a) the occupied bandwidth, b) the encompassed spectrum, c) the modulated spectrum, and d) the number of equivalent legacy DOCSIS channels.

The encompassed spectrum in MHz is 204.8 MHz minus the number of subcarriers in the Band edge Exclusion Sub-band for the upper and lower band edges (combined) times the subcarrier spacing in MHz. For example, with subcarrier spacing of 50 kHz and 150 lower band edge subcarriers and 152 upper band edge subcarriers for a total of 302 subcarriers in the two Band edge Exclusion Sub-bands, the encompassed spectrum = 204.8 – 302*(0.05) = 189.7 MHz. The encompassed spectrum is also equal to the center frequency of the highest frequency modulated subcarrier minus the center frequency of the lowest frequency modulated subcarrier in an OFDM channel, plus the subcarrier spacing.

The modulated spectrum of an OFDM channel is the encompassed spectrum minus the total spectrum in the Internal Excluded Sub-bands of the channel, where the total spectrum in the Internal Excluded Sub-bands is equal to the number of subcarriers in all of the Internal Excluded Sub-bands of the OFDM channel multiplied by the subcarrier spacing of the OFDM channel. In the previous example, if there are 188 subcarriers total in three Internal Exclusion Sub-bands, then the total spectrum in the Internal Excluded Sub-bands (in MHz) is 188*0.05 = 9.4 MHz, and the modulated spectrum is 189.7 MHz – 9.4 MHz = 180.3 MHz.

The occupied bandwidth is a multiple of 6 MHz, with a minimum of 24 MHz, and consists of all CTA channels which include the modulated spectrum plus taper region shaped by the OFDM channels' transmit windowing; the out-of-band spurious emissions requirements apply outside the occupied bandwidth. With a 1 MHz taper region on each band edge of the OFDM channel, shaped by the transmit windowing function, encompassed spectrum of 189.7 MHz may provide 192 MHz of occupied bandwidth.

The number of equivalent active legacy DOCSIS channels in the OFDM channel Neq' is the ceiling function applied to the modulated spectrum divided by 6 MHz. For the example, the number of equivalent legacy DOCSIS channels in the OFDM channel is ceiling(180.3 MHz / 6 MHz) = 31.

For an Neq-channel per RF port device, the applicable maximum power per channel and spurious emissions requirements are defined using a value of N* = minimum(4Neq', ceiling[Neq/4]) for Neq' < Neq/4, and N* = Neq' otherwise.

These specifications assume that the RPD will be terminated with a 75 Ohm load.

F.2.1 RPD Output Electrical Requirements

The requirements contained in DOCSIS 3.1 PHY Section 7.5.9.1 CMTS Output Electrical Requirements apply for Remote PHY Devices, except for the following changes:

- "RPD" replaces "CMTS" in all instances.
- Reference to "Table 7-37" is replaced with reference to "Table F-1" in all instances.
- The sentence in the second paragraph, "Legacy DOCSIS RF modulated signal characteristics are provided in Section 6.2.22," is replaced with, "Legacy DOCSIS RF modulated signal characteristics for the RPD are provided in DRFI Annex D."
- In Table 7-36, row "Level," the right-column entry "Adjustable. See Table 7-37," is replaced by "Adjustable. See Table F-1."
- In Table 7-36, row "Connector", the right-column entry becomes "F connector per ISO/IEC-61169-24 or [SCTE 02], or 75 ohm MCX [ANSI/SCTE 176]table_note_A, or 75 ohm SMB [MIL-STD-348]table_note_B."
- The following two table notes are added to Table 7-36: "Table note_A. CCAP spec approved and commonly used in CMTS/EQAM," and "Table note_B. Commonly used in nodes."

F.2.1.1 *Power per Channel for RPD*

[This section replaces DOCSIS 3.1 PHY Section 7.5.9.1.1 Power per Channel for CMTS, for Remote PHY Devices.]

Remote PHY Devices (RPDs) perform the modulation of channels which are ordinarily generated by EQAMs and CMTSs at the headend, which are defined in the main section of [PHYv3.1].

Control over an RPD's electrical output is required for many of the characteristics, such as RF channel power, number of RF channels, modulation characteristics of the channels, center frequency of channels, and so forth. Two distinct mechanisms of control can exist for an RPD. One mechanism of control is via commands carried in the downstream link into the RPD. A second mechanism of control is "local-only", separate from the downstream link into the RPD, such as an electrical interface operable at installation or even pluggable components set at installation. In an RPD some adjustable characteristics can be controlled by one mechanism, and not the other, or by both; therefore, some "adjustable" characteristics can perhaps not be remotely changed. Local-only adjustments made at installation can be subsequently amended, but not remotely, and could incur service interruption.

An RPD is capable of generating some maximum number of equivalent legacy DOCSIS channels onto the RF port, Neq, and is capable of generating a power per channel of at least 20 dBmV/6 MHz. The Channel Power Reference Setting (dBmV/6 MHz) of the RPD could possibly be adjustable remotely, but is also permitted to be adjustable only locally, or even fixed (not adjustable), and serves as the reference power (0 dBc) for independently controlled individual channel power adjustment, and for spurious emissions. The power per channel of the RPD has to be adjustable for each channel independently, via remote adjustment, over a range of 2 dB below the Channel Power Reference Setting. An RPD has to be adjustable to operate with fewer than Neq-channels on its RF port. An Neq-channel per RF port device has to comply with all requirements operating with all Neq channels on the RF port, and has to comply with all requirements for an Neq'-channel per RF port device operating with Neq' channels on the RF port for all values of Neq' less than Neq that it supports.

For an Neq-channel per RF port device with Neq' < Neq/4, the applicable spurious emissions requirements are defined using a value of N* = minimum(4Neq', ceiling[Neq/4]).

An Neq-channel per RF port RPD MUST support operation over Neq ≥ Neq' ≥ Neq/4. The RPD MUST maintain the commanded power when operating with any Neq' over this range. The RPD MUST meet Tables 7-36 [PHYv3.1], with changes as described in Section F.2.1, Table F-1, and Table 7-38 in [PHYv3.1], when operating with any Neq' over this range.

An Neq-channel per RF port RPD SHOULD support operation over Neq/4 > Neq' ≥ Neq/16. The RPD SHOULD maintain the commanded power when operating with any Neq' over this range. The RPD MUST meet Tables 7-36 [PHYv3.1], with changes as described in Section F.2.1, Table F-1, and Table 7-38 in [DOCSIS 3.1 PHY], when operating with any Neq' over this range.

An Neq-channel per RF port RPD MAY support operation over Neq' < Neq/16. The RPD SHOULD maintain the commanded power when operating with any Neq' over this range. The RPD MUST meet Tables 7-36 [DOCSIS 3.1 PHY], with changes as described in Section F.2.1, Table F-1, and Table 7-38 in [PHYv3.1], when operating with any Neq' over this range.

These specifications assume that the RPD device will be terminated with a 75 ohm load.

An RPD MUST generate an RF output with power capabilities as defined in Table F-1.

The RPD MUST be capable of adjusting channel RF power on a per channel basis as stated in Table F-1.

The RPD MUST be capable of adjusting power on a per channel basis for the legacy DOCSIS channels, with each channel independently meeting the power capabilities defined in Table 29.

Table 29 - RPD Output Power

$\textbf{for } N^* \equiv \begin{cases} minimum[4N_{eq}', \text{ceiling}\left[\frac{N_{eq}}{4}\right]], & N_{eq}' < N_{eq}/4 \\ N_{eq}', & N_{eq}' \geq N_{eq}/4 \end{cases}$, **Adjusted Number of Active Channels Combined per RF Port**	
Parameter	**Value**
Channel Power Reference Setting (Maximum required power, i.e., 0 dBc, per channel for N_{eq}' channels combined onto a single RF port for an N_{eq} channel RPD):	Required power in dBmV per 6 MHz channel must: ≥ 20 dBmV NOTES: No upper limit to the Channel Power Reference Setting which an RPD can provide No requirement for remote adjustability for Channel Power Reference Setting An RPD which has fixed Channel Power Reference Setting must meet full fidelity (Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per Section F.2.1, and Table 7-38 in [PHYv3.1]) at that setting. For RPD which has adjustable Channel Power Reference Setting, see the corresponding row in this table.
Range of Channel Power Reference Setting	may be adjustable locally An RPD with adjustable Channel Power Reference Setting must meet full fidelity (Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per Section F.2.1, and Table 7-38 in [PHYv3.1]) whenever Channel Power Reference Setting is at or below: 60 – ceil [3.6*log2(Neq)] dBmV For RPD with Channel Power Reference Setting range which does not reach as low as 60 – ceil [3.6*log2(Neq)] dBmV, the RPD must meet full fidelity with Channel Power Reference Setting at the lowest setting for the device.

for $N^* \equiv \begin{cases} minimum[4N_{eq}', \text{ceiling}\left[\frac{N_{eq}}{4}\right]], & N_{eq}' < N_{eq}/4 \\ N_{eq}', & N_{eq}' \geq N_{eq}/4 \end{cases}$, Adjusted Number of Active Channels Combined per RF Port	
Parameter	**Value**
Range of commanded power per channel; adjusted on a per channel basis	CMTS must: 0 dBc to -2 dBc relative to Channel Power Reference Setting, via remote adjustment. may: larger variations than 2 dB below Channel Power Reference Setting, via remote adjustment.
Commanded power per channel step size	≤ 0.2 dB Strictly monotonic
Power difference between any two adjacent channels in the 108-1218 MHz downstream spectrum (with commanded power difference removed if channel power is independently adjustable and/or accounting for pilot density variation and subcarrier exclusions)	≤ 0.5 dB
Power difference between any two non-adjacent channels in a 48 MHz contiguous bandwidth block (with commanded power difference removed if channel power is independently adjustable)	≤ 1 dB
Power difference (normalized for bandwidth) between any two channels OFDM channel blocks or legacy DOCSIS channels in the 108 - 1218 MHz downstream spectrum (with commanded power difference removed if channel power is independently adjustable)	≤ 2 dB
Power per channel absolute accuracy	±3 dB Table footnote: This specification contains a stability requirement which is much tighter than +-3 dB.
Diagnostic carrier suppression (3 modes) Mode 1: One channel suppressed must be controlled remotely Mode 2: All channels suppressed except one must be controlled remotely Mode 3: All channels suppressed must be controlled remotely	1) ≥ 50 dB carrier suppression within the occupied bandwidth in any one active channel. When suppressing the carrier ≥ 50 dB within the occupied bandwidth in any one active channel the CMTS must control transmissions such that no service impacting discontinuity or detriment to the unsuppressed channels occurs. 2) 50 dB carrier suppression within the occupied bandwidth in every active channel except one. The suppression is not required to be glitchless, and the remaining unsuppressed active channel is allowed to operate with increased power such as the total power of the N_{eq}' active channels combined. 3) 50 dB carrier suppression within the occupied bandwidth in every active channel. The CMTS must control transmissions such that in all three diagnostic carrier suppression modes the output return loss of the suppressed channel(s) complies with the Output Return Loss requirements for active channels given in Table 7-36 RF Output Electrical Requirements[PHYv3.1] as amended per Section F.2.1. The total noise and spur requirement is the combination of noise power from the 50 dBc suppressed channel and the normal noise and spur requirement for the CMTS output when operating with all channels unsuppressed.

for $N^* \equiv \begin{cases} minimum[4N_{eq}', \text{ceiling}\left[\frac{N_{eq}}{4}\right]], & N_{eq}' < N_{eq}/4 \\ N_{eq}', & N_{eq}' \geq N_{eq}/4 \end{cases}$, Adjusted Number of Active Channels Combined per RF Port	
Parameter	**Value**
RF output port muting	≥ 73 dB below the unmuted aggregate power of the RF modulated signal, in every 6 MHz CTA channel from 54 MHz to 1218 MHz. The specified limit applies with all active channels commanded to the same transmit power level. Commanding a reduction in the transmit level of any, or all but one, of the active channels does not change the specified limit for measured muted power in 6 MHz. When the CMTS is configured to mute an RF output port, the CMTS must control transmissions such that the output return loss of the output port of the muted device complies with the Output Return Loss requirements for inactive channels given in Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per Section F.2.1.
Table Notes Note 1: "Channel" in mode 1 or mode 2 carrier suppression refers to an OFDM channel with at least 22 MHz of contiguous modulated spectrum or an SC-QAM channel.	

The following is a list of RF Output Electrical Requirements based on Table 29 above.

- In an RPD, Channel Power Reference Setting MUST be ≥ 20 dBmV
  - An RPD which has fixed Channel Power Reference Setting MUST meet full fidelity (Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per Section F.2.1, and Table 7-38 [PHYv3.1]) at that setting.
- In an RPD, Range of Channel Power Reference Setting MAY be adjustable locally
  - An RPD with adjustable Channel Power Reference Setting MUST meet full fidelity (Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per Section F.2.1, and Table 7-38 [PHYv3.1]) whenever Channel Power Reference Setting is at or below: 60 – ceil [3.6*log2(Neq)] dBmV.
  - For RPD with Channel Power Reference Setting range which does not reach as low as 60 – ceil [3.6*log2(Neq)] dBmV, the RPD MUST meet full fidelity with Channel Power Reference Setting at the lowest setting for the device.
- In an RPD, the range of commanded power per channel, adjusted on a per channel basis, MAY be according to the larger variations than 2 dB below Channel Power Reference Setting, via remote adjustment.
- In an RPD, the range of commanded power per channel, adjusted on a per channel basis, a CMTS MUST be 0 dBc to -2 dBc relative to Channel Power Reference Setting, via remote adjustment.
- In an RPD, Diagnostic carrier suppression (3 modes)
  - Mode 1: One channel suppressed to be controlled remotely. For this mode, the CMTS MUST support ≥ 50 dB carrier suppression within the occupied bandwidth in any one active channel. When suppressing the carrier ≥ 50 dB within the occupied bandwidth in any one active channel, the CMTS MUST control transmissions such that no service impacting discontinuity or detriment to the unsuppressed channels occurs.
  - Mode 2: All channels suppressed except one to be controlled remotely. For this mode, the CMTS MUST support 50 dB carrier suppression within the occupied bandwidth in any one active channel except one. The suppression is not required to be glitchless, and the remaining unsuppressed active channel is allowed to operate with increased power such as the total power of the N_{eq}' active channels combined.
  - Mode 3: All channels suppressed to be controlled remotely. For this mode, the CMTS MUST support 50 dB carrier suppression within the occupied bandwidth in every active channel.

- The CMTS MUST control transmissions such that in all three diagnostic carrier suppression modes, the output return loss of the suppressed channel(s) complies with the Output Return Loss requirements for active channels given in Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per Section F.2.1.
- When the CMTS is configured to mute an RF output port, the CMTS MUST control transmissions such that the output return loss of the output port of the muted device complies with the Output Return Loss requirements for inactive channels given in Table 7-36 RF Output Electrical Requirements [PHYv3.1] as amended per section F.2.1.

F.2.1.2 *Out-of-Band Noise and Spurious Requirements for RPD*

The requirements contained in DOCSIS 3.1 PHY Out-of-Band Noise and Spurious Requirements for CMTS, apply for Remote PHY Devices, except "RPD" replaces "CMTS" in all instances and reference to "Table 7-37" is replaced with reference to "Table F-1" in all instances.

Appendix I Plant Sweep in a Distributed Architecture (Informative)

Today, operators in HFC plants deploy test equipment that allows sweep tests to be performed, measuring plant frequency response in the upstream and downstream direction. Traditionally, these have been closed, proprietary systems with these characteristics:

- In the downstream, proprietary equipment in the plant generates sweep signals that are measured by field test equipment; a control channel between the headend equipment and test equipment controls how and when these signals are generated.
- In the upstream, the test equipment in the field generates signals that are measured by proprietary equipment in the headend; a similar control channel between the test equipment and headend equipment is used to feed measurements back to the test equipment so that adjustments can be made.

In a Remote PHY architecture, supporting the telemetry/control channel between the headend and the field test equipment becomes a challenge. In a traditional architecture, the headend equipment is connected through the combining network; this connection is eliminated in the R-PHY architecture. Other methods for performing sweep are needed.

In this appendix, three alternatives to using currently available plant maintenance systems are discussed:

- Using current transmitter and receiver technology, developed as part of the DOCSIS Proactive Network Maintenance (PNM) toolset, to perform measurements;
- Introducing modules to the R-PHY Node that perform the role of the headend test equipment;
- Developing an API in the R-PHY Node that allows interaction with field test equipment.

I.1 Plant Sweep Using Transmitter and Receiver Capabilities

With the full-band capture capabilities introduced for DOCSIS 3.0 and 3.1 equipment, frequency response measurements can be taken by either the CM or the R-PHY node receiver. Existing signals in the plant can be used in the downstream for these measurements and the results of the measurements can be made available to test equipment in the field via SNMP. To measure portions of the spectrum where no signals exist (for example, when evaluating regions where services will be expanded for DOCSIS 3.1), the CCAP Core can instruct the R-PHY node to generate signals that can be measured by the CM.

In the upstream, existing signals can be measured and test modes on the CM can generate carrier signals that can be measured at the R-PHY Node burst receiver. These measurements too can be exposed to field test equipment via SNMP.

In addition, PNM enables symbol capture in both the upstream and downstream direction, allowing impairments to also be detected in the time domain, rather than just the frequency domain.

Details on the PNM toolset can be found in the following DOCSIS 3.1 specifications: [CCAP-OSSIv3.1], [CM-OSSIv3.1], [MULPIv3.1], and [PHYv3.1].

I.2 Hardware Module in the Node

Test equipment vendors may develop modules that will be deployed within a node that supports the R-PHY architecture that performs the same function as the equipment that was previously deployed in the headend. Since the module is located in the R-PHY Node, the same telemetry and control channels can be used. In this approach, the sweep vendors can work with the node vendors to develop the sweep module and therefore the topic is not covered in detail in this specification.

I.3 R-PHY Node API Support

In this approach, an API is developed by R-PHY Node and test equipment vendors that can be used by test equipment to control the placement and configuration of signals in the RF spectrum. This API provides more control of sweep carrier generation and access to measurements by the test equipment, without the need to support a specific hardware module in the node, as described in the previous approach. Since the sweep signal itself is a CW signal, no

additional RF capability is required above what is defined in the R-PHY specifications (i.e., the ability to generate CW carriers at any frequency and the ability to measure RF receive levels).

Appendix II Acknowledgements

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
John T. Chapman	Cisco
Pawel Sowinski	Cisco
Gerry White	Cisco
Stuart Hoggan	CableLabs
Michael Patrick	Harmonic

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 the technology and participation in the Remote PHY Working Group.

Contributor	Company Affiliation
Bill Powell	Alcatel-Lucent
Brian Kurtz	Altera
Carlton Lane, Linda Mazaheri	Analog
Tom Ferreira, Steve Foley, Anand Goenka, Jeff Howe, Hari Nair	Arris
Andrew Chagnon, Victor Hou, Niki Pantelias, David Pullen	Broadcom
Stuart Hoggan, Volker Leisse, Jon Schnoor, Karthik Sundaresan, Nikhil Tayal, Jun Tian	CableLabs
Andrew Sundelin	CableLabs Consultant
Naor Goldman	Capacicom
Dave Fox, Maike Geng	Casa Systems
David Claussen	Charter
Nobo Akiya, Alon Bernstein, Brian Bresnahan, John T. Chapman, Hang Jin, Tong Liu, Carlos Pignataro, Sangeeta Ramakrishnan, John Ritchie, Pawel Sowinski, Don Strausberger, Yi Tang, Xiaoming (Shaun) Yu, Bill Wall, Gerry White	Cisco
Philippe Perron	Cogeco
John Bevilacqua, Nagesh Nandiraju, Saifur Rahman, Jorge Salinger, Joe Solomon, Douglas Will	Comcast
Jeff Ford, Al Garrett	Complex IQ
Ony Anglade, Mike Cooper	Cox Communications
Samir Parikh	Gainspeed Networks
João Campos, Even Kristoffersen	Get
Adi Bonen, Mike Patrick	Harmonic
Jim Chen, Hesham ElBakoury, Karl Moerder, Jack Moran, Guangsheng Wu	Huawei
Phil Oakley	LGI
Stan Bochenek, Ajay Kuckreja	Maxim Integrated
Len Dauphinee, David Huang, Louis Park, Sridhar Ramesh, Patrick Tierney, Scott Walley	MaxLinear
Rei Brockett	Pace/Aurora
Nasir Ansari, George Hart	Rogers
Kevin Kwasny	Shaw
Lee Johnson	ST Micro
Paul Brooks, Kirk Erichsen	Time Warner Cable
Colin Howlett, Douglas Johnson	Vecima
Faten Hijazi, Alex Luccisano	Xilinx

Additionally, CableLabs would like to thank the DCA MSO team for their continued support in driving the specification development and the decision-making process.

Karthik Sundaresan, CableLabs

Appendix III Revision History

III.1 Engineering Changes for CM-SP-R-PHY-I02-151001

ECN	Accepted	Summary	Author
R-PHY-N-15.1357-2	09/09/2015	Hold RPD Boot To Allow Debug	White
R-PHY-N-15.1359-1	09/09/2015	Pilot tones and alignment carriers	Sowinski
R-PHY-N-15.1360-4	09/09/2015	GCP TLV Encoding	Sowinski

III.2 Engineering Changes for CM-SP-R-PHY-I03-160121

ECN	Accepted	Summary	Author
R-PHY-N-15.1401-1	12/2/2015	Downstream Symbol Capture and Upstream Histogram in R-PHY	Sowinski
R-PHY-N-15.1403-3	12/16/2015	GCP Protocol Definition	Sowinski
R-PHY-N-15.1410-2	12/16/2015	Updates to RPD Secure Software Download for R-PHY	Sowinski

III.3 Engineering Changes for CM-SP-R-PHY-I04-160512

ECN	Accepted	Summary	Author
R-PHY-N-16.1444-1	3/17/2016	Remove duplicate text	Sowinski
R-PHY-N-16.1451-1	4/21/2016	GCP encoding for configuration of 55-2 modules and SID QoS.	Sowinski
R-PHY-N-16.1476-1	4/21/2016	TLVs for RPD Monitoring	Cookish
R-PHY-N-16.1477-2	4/21/2016	NDF and NDF Configuration, Multiple GCP encoding issues	Sowinski
R-PHY-N-16.1487-2	4/21/2016	Move pilot tones and plant sweep appendix to R-OOB	Bonen

III.4 Engineering Changes for CM-SP-R-PHY-I05-160923

ECN	Accepted	Summary	Author
R-PHY-N-16.1514-1	6/2/2016	Certificate PKI Profile Updates	Hoggan
R-PHY-N-16.1545-2	6/30/2016	GCP Configuration of SCTE 55-1 OOB channels.	Sowinski
R-PHY-N-16.1551-1	7/28/2016	Discrepancies of specs in R-PHY-I04 versus R-DEPI-I04 for MCM and DMPT for DOCSIS (RPHY-161)	Huang
R-PHY-N-16.1556-3	8/4/2016	RPHY - NumSymbolsPerFrame fix	Schnoor
R-PHY-N-16.1562-1	8/4/2016	Supersedes ECN 1545: GCP Configuration of SCTE 55-1 OOB channels	Sowinski
R-PHY-N-16.1564-2	8/18/2016	RPD PTP Slave Configuration TLVs for G.8275.2 profile.	Sowinski
R-PHY-N-16.1570-1	8/18/2016	Update abbreviation table with RCP	Schnoor
R-PHY-N-16.1573-3	9/1/2016	Clarification of MAC Management Message use in R-PHY configuration protocol.	Sowinski
R-PHY-N-16.1575-1	9/1/2016	RPHY editorial MUST statement fixes	Schnoor
R-PHY-N-16.1576-1	9/1/2016	GDC TLVs for support of downstream buffer monitoring and buffer depth alerts.	Sowinski
R-PHY-N-16.1584-2	9/1/2016	RPHY Internal Components diagram fix	ElBakoury
R-PHY-N-16.1586-2	9/1/2016	Remove RPD Monitoring GCP TLVs from R-PHY spec	Patrick

III.5 Engineering Changes for CM-SP-R-PHY-I06-170111

ECN	Accepted	Summary	Author
R-PHY-N-16.1616-1	10/20/2016	RPHY Annex E addition	Bonen
R-PHY-N-16.1637-1	11/17/2016	RPHY Update for DHCP suboption 61	Schnoor
R-PHY-N-16.1644-3	12/15/2016	Annex B omnibus	Schnoor
R-PHY-N-16.1663-1	12/15/2016	D3.1 OFDM Modifications for RPHY (Annex F)	Kolze
R-PHY-N-16.1673-1	12/15/2016	Typo corrections in R-PHY	Egorov