

Superseded

Data-Over-Cable Service Interface Specifications

eDOCSIS™ Specification

SP-eDOCSIS-I02-031117

ISSUED

Notice

This specification is a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc. (CableLabs®) for the benefit of the cable industry. Neither CableLabs, nor any other entity participating in the creation of this document, is responsible for any liability of any nature whatsoever resulting from or arising out of use or reliance upon this document by any party. This document is furnished on an AS-IS basis and neither CableLabs, nor other participating entity, provides any representation or warranty, express or implied, regarding its accuracy, completeness, or fitness for a particular purpose.

© Copyright 2003 Cable Television Laboratories, Inc.
All rights reserved.

Document Status Sheet

Document Control Number:	SP-eDOCSIS-I02-031117			
Document Title:	Data-Over-Cable Service Interface Specifications eDOCSIS™ Specification			
Revision History:	I01 - Issued March 12, 2003			
	I02 - Issued November 17, 2003			
Date:	November 17, 2003			
Status:	Work in Progress	Draft	Issued	Closed
Distribution Restrictions:	Author Only	CL/Member	CL/Member/Vendor	Public

Key to Document Status Codes:

- Work in Progress** An incomplete document, designed to guide discussion and generate feedback, that may include several alternative requirements for consideration.
- Draft** A document in specification format considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
- Issued** A stable document, which has undergone rigorous member and vendor review and is suitable for product design and development, cross-vendor interoperability, and for certification testing.
- Closed** A static document, reviewed, tested, validated, and closed to further engineering change requests to the specification through CableLabs.

Trademarks:

DOCSIS®, eDOCSIS™, PacketCable™, CableHome™, OpenCable™, CableCARD™, and CableLabs® are trademarks of Cable Television Laboratories, Inc.

Contents

1	INTRODUCTION	1
1.1	Scope.....	1
1.2	Goals	1
1.3	DOCSIS Base Specifications.....	1
1.4	Requirements	2
2	REFERENCES	3
2.1	References (normative)	3
2.2	References (informative)	4
2.3	Reference Acquisition	4
3	GLOSSARY	5
4	ABBREVIATIONS	6
5	EMBEDDED DOCSIS CABLE MODEM	7
5.1	Device Interface Reference Model	7
5.2	eDOCSIS Requirements.....	9
5.2.1	General Requirements	9
5.2.2	Interface Requirements	10
5.2.3	Operations Support Requirements	10
5.2.4	DHCP Option 43 Syntax Requirements	13
5.2.5	Testability Requirements	18
	ANNEX A SLED MIB DEFINITION	24
	APPENDIX A ACKNOWLEDGEMENTS.....	29
	APPENDIX B REVISION HISTORY	30

Figures

Figure 5-1 - eDOCSIS Reference Model	7
Figure 5-2 - CableHome Home Access eDOCSIS Device Reference Model	8
Figure 5-3 - eCM - ePS Protocol Stacks	8
Figure 5-4 - PacketCable E-MTA (with DOCSIS CM) eDOCSIS Reference Model.....	9
Figure 5-5 - eCM - eMTA Protocol Stacks	9
Figure 5-6 - SLED Reference Model.....	19
Figure 5-7 - SLED Packet Loopback Encapsulation	21
Figure 5-8 - SLED Packet Loopback And Generation Sequences	23

Tables

Table 5-1 - eDOCSIS ifTable Interface Designations	11
Table 5-2 - [RFC 2863] ifTable, MIB-Object Details for eDOCSIS Device Interfaces...	12
Table 5-3 - [RFC 2011] ipNetToMedia MIB-Object Details for eDOCSIS Device Interfaces	13
Table 5-4 - DHCP Option 43 Syntax	15
Table 5-5 - Example eDOCSIS Device: E-MTA.....	16
Table 5-6 - Example eDOCSIS Device: DOCSIS 1.1 CableHome/E-PS	16
Table 5-7 - Example eDOCSIS Device Containing DOCSIS 2.0 eCM, eMTA and ePS 18	

1.1 **Superseded**

Existing DOCSIS specifications were created for stand-alone cable modems that provide high-speed broadband services using the hybrid-fiber-coaxial cable infrastructure. The emergence of a class of devices that embeds additional functionality with a Cable Modem, such as packet-telephony, home networking and video, has necessitated the creation of this specification to define additional requirements such as interfaces, management and provisioning models. This is necessary to insure that the Cable Modem will function properly and interact properly with the embedded Service/Application Functional Entities (eSAFEs).

1.1 Scope

This specification defines additional features that must be added to a DOCSIS CableModem for implementations that embed the Cable Modem with another application, such as a PacketCable MTA.¹

1.2 Goals

The goals for this specification are:

- To preserve functional separation of the DOCSIS cable modem entity from eSAFEs within the eDOCSIS Device, so that existing DOCSIS cable plant integrity, cable modem configuration, management and provisioning security are not compromised.
- To isolate DOCSIS cable modem functionality so that specification compliance can be tested for the eCM component independent of eSAFEs.
- To enable the service provider to enable or disable forwarding traffic between each eSAFE and the eCM within the eDOCSIS Device.
- To maximize compatibility with existing back-office management/provisioning infrastructure so that new services enabled by eDOCSIS devices can be deployed rapidly.
- To architect eDOCSIS devices in such a way as to scale to new services and applications, and to take advantage of technology innovations to achieve low cost and high functionalities.²

1.3 DOCSIS Base Specifications

There are currently three versions of what are in this document referred to as the DOCSIS Base Specifications. These three versions are commonly referred to as DOCSIS 1.0, DOCSIS 1.1, and DOCSIS 2.0. A list of the document categories in the Data-Over-Cable Service Interface Specifications family is provided below. For updates, please refer to <http://www.cablemodem.com/>.

¹ Edited per EDOCS-N-03099, 11/11/03, po

² Edited per EDOCS-N-03099, 11/11/03, po

Designation			Title
DOCSIS 1.0	DOCSIS 1.1	DOCSIS 2.0	
SP-RFI	SP-RFIv1.1	SP-RFIv2.0	Radio Frequency Interface Specification
SP-OSSI	SP-OSSIv1.1	SP-OSSIv2.0	Operations Support System Interface Specification
SP-BPI	SP-BPI+		Baseline Privacy Interface Specification
SP-CMCI			Cable Modem to Customer Premises Equipment Interface Specification
SP-CMTS-NSI			Cable Modem Termination System Network Side Interface Specification

1.4 Requirements

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

MUST	This word or the adjective “REQUIRED” 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 or the adjective “RECOMMENDED” 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 or the adjective “OPTIONAL” 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

2.1 References (normative)

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.

- [DOCSIS0-CMCI] “Cable Modem to Customer Premise Equipment Interface Specification, SP-CMCI-I09-030730”, July 30, 2003, CableLabs.
- [RFI 1.0] SCTE 22-1 2002: DOCSIS 1.0 Radio Frequency Interface”, www.scte.org.
- [RFI 1.1] “Radio Frequency Interface Specification, SP-RFIV1.1-I10-030730”, July 30, 2003, CableLabs.
- [RFI 2.0] “Radio Frequency Interface Specification, SP-RFIV2.0-I04-030730”, July 30, 2003, CableLabs.
- [OSSI 1.0] SCTE 22-3 2002: “DOCSIS 1.0 Operations Support System Interface”, www.scte.org.
- [OSSI 1.1] “Operations Support System Interface Specification, SP-OSSIV1.1-I07-030730”, July 30, 2003, CableLabs.
- [OSSI 2.0] “Operations Support System Interface Specification, SP-OSSIV2.0-I04-030730”, July 30, 2003, CableLabs.
- [RFC 1493] IETF RFC 1493, Definitions of Managed Objects for Bridges, E. Decker, P. Langille, A. Rijsinghani & K. McCloghrie, July 1993.
- [RFC 2011] IETF RFC 2011, SNMPv2 Management Information Base for the Internet Protocol using SMIV2, K. McCloghrie, November 1996.
- [RFC 2863] IETF RFC 2863, The Interfaces Group MIB, K. McCloghrie, F. Kastenholz, June 2000.
- [RFC 2131] IETF RFC 2131, Dynamic Host Configuration Protocol, Droms, R., March 1997.
- [RFC 2132] IETF RFC 2132, DHCP Options and BOOTP Vendor Extensions, Alexander, S., and R. Droms, March 1997.
- [RFC 3396] IETF RFC 3396, Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4), Lemon, T., and S. Cheshire, November, 2002.³
- [RFC 791] IETF STD5, RFC 791, Internet Protocol, J. Postel, September 1981.
- [RFC 768] IETF STD6, RFC 768, User Datagram Protogram, J. Postel, September, 1980.

³ Edited per EDOCS-N-03084/ 11/11/03, po

2.2 References (informative)

- [CH10] “CableHome 1.0 Specification SP-CH1.0-I05-030801”, August 1, 2003, CableLabs.
- [PC10-MTA] “PacketCable MTA Device Provisioning Specification, PKTC-SP-PROV-I07-030728, July 28, 2003, CableLabs.
- [PC10-MIB] “PacketCable MIBs Framework Specification, PKTC-SP-MIBS-I07-030728”, July 28, 2003, CableLabs.
- [CH11] “CableHome 1.1 Specification, CH-SP-I02-030801”, August 1, 2003, CableLabs.⁴

2.3 Reference Acquisition

CableLabs Specifications:

- Cable Television Laboratories, Inc., 400 Centennial Parkway, Louisville, CO 80027; Phone 303-661-9100; Fax 303-661-9199, www.cablelabs.com
- SCTE Standards, Society of Cable Telecommunication Engineers, 140 Philips Road, Exton PA 19341-1318, Phone 610-524-1725, www.scte.org
- IETF Standards, Internet Engineering Task Force (IETF) Secretariat, c/o Corporation for National Research Initiatives, 1895 Preston White Drive, Suite 100, Reston, VA 20191-5434, Phone 703-620-8990, Fax 703-620-9071, www.ietf.org

⁴ Edited per EDOCS-N-03084/ 11/11/03, po

3 GLOSSARY

CMCI	Cable Modem (CM) to Customer Premise Equipment (CPE) Interface as defined in [DOCSIS0-CMCI].
eCM	An eCM is an embedded Cable Modem, i.e., one that has been enhanced with the features of this specification.
eDOCSIS	eDOCSIS is the embedded DOCSIS specification that defines the interface between the eCM and an eSAFE.
eDOCSIS Device	An eDOCSIS Device is one that includes an eCM entity, one or more eSAFEs and supports a single software image using a DOCSIS secured software download mechanism. ⁵
eMTA	Embedded Multimedia Terminal Adaptor. An embedded version of an MTA.
E-MTA	Embedded MTA device. An eDOCSIS device that contains both an eMTA and an eCM.
ePS	Embedded Portal Service Element. A CableHome-compliant eSAFE that provides management and network address translation functions between the DOCSIS network and the home network.
Logical CPE Interface (Logical CMCI)	A bi-directional, data-only 802.3/Ethernet MAC frame interface between eCM and an eSAFE.
MTA	Multimedia Terminal Adapter as defined in [PC10-MTA]. Contains the interface to a physical voice device, a network interface, CODECs, and all signaling and encapsulation functions required for VoIP transport, class features signaling and QoS signaling.
eSAFE (embedded Service/Application Functional Entity)	An embedded version of CableLabs-specified application, such as a PacketCable Multimedia Terminal Adapter (MTA), that provides a service using the DOCSIS IP platform, or a function or set of functions, such as the CableHome Portal Services logical element, that supports the delivery of one or more services over an IP platform.

⁵ Edited per EDOCS-N-03099, 11/11/03, po

4 ABBREVIATIONS

This specification uses the following abbreviations:

CM	Cable Modem
CMCI	Cable Modem to Customer Premises Equipment Interface
DOCSIS	Data-Over-Cable Service Interface Specifications
eCM	Embedded Cable Modem
eDOCSIS	Embedded DOCSIS
eMTA	Embedded MTA
ePS	Embedded Portal Services Element
eSAFE	Embedded Service/Application Functional Entity

5 EMBEDDED DOCSIS CABLE MODEM

5.1 Device Interface Reference Model

Referring to Figure 5-1, an eDOCSIS device consists of an embedded DOCSIS cable modem (eCM) and one or more embedded Service/Application Functional Entities (eSAFEs). An eDOCSIS device may also have one or more physically exposed interfaces. In addition, only a single secured software image download is used for the entire eDOCSIS device.

eSAFEs include:

- ePS: embedded CableHome Portal Services Logical Element [CH10]
- eMTA: embedded PacketCable Multimedia Terminal Adapter [PC10-MTA, PC10-MIB]

Within an eDOCSIS device, each eSAFE interfaces to the eCM via a point-to-point logical CPE interface.

Figure 5-2 presents a typical CableHome Home Access eDOCSIS Device reference model.

Figure 5-3 presents a logical view of protocol stacks for an eCM to ePS interface.

Figure 5-4 presents a typical PacketCable E-MTA (with DOCSIS cable modem) eDOCSIS Device reference model.

Figure 5-5 presents a logical view of protocol stacks for an eCM to eMTA interface.

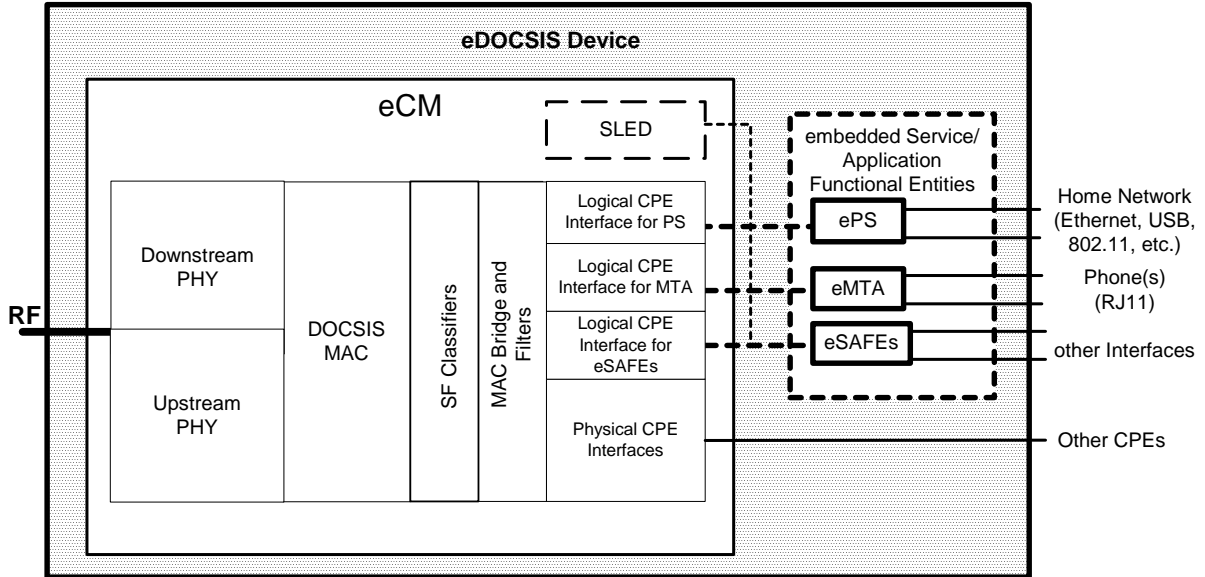


Figure 5-1 - eDOCSIS Reference Model

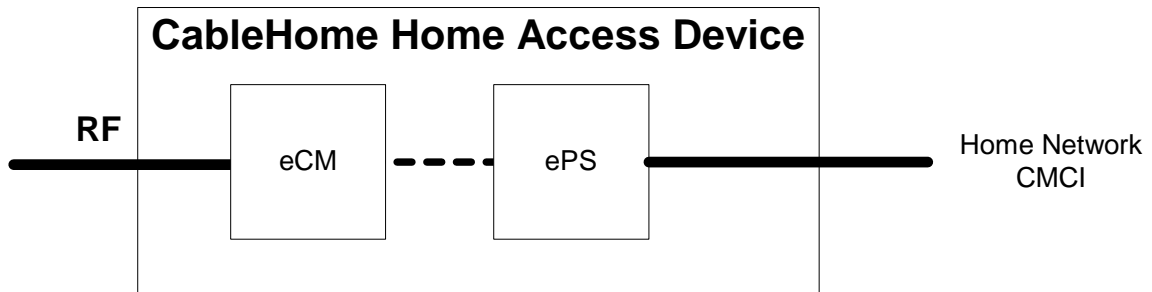


Figure 5-2 - CableHome Home Access eDOCSIS Device Reference Model

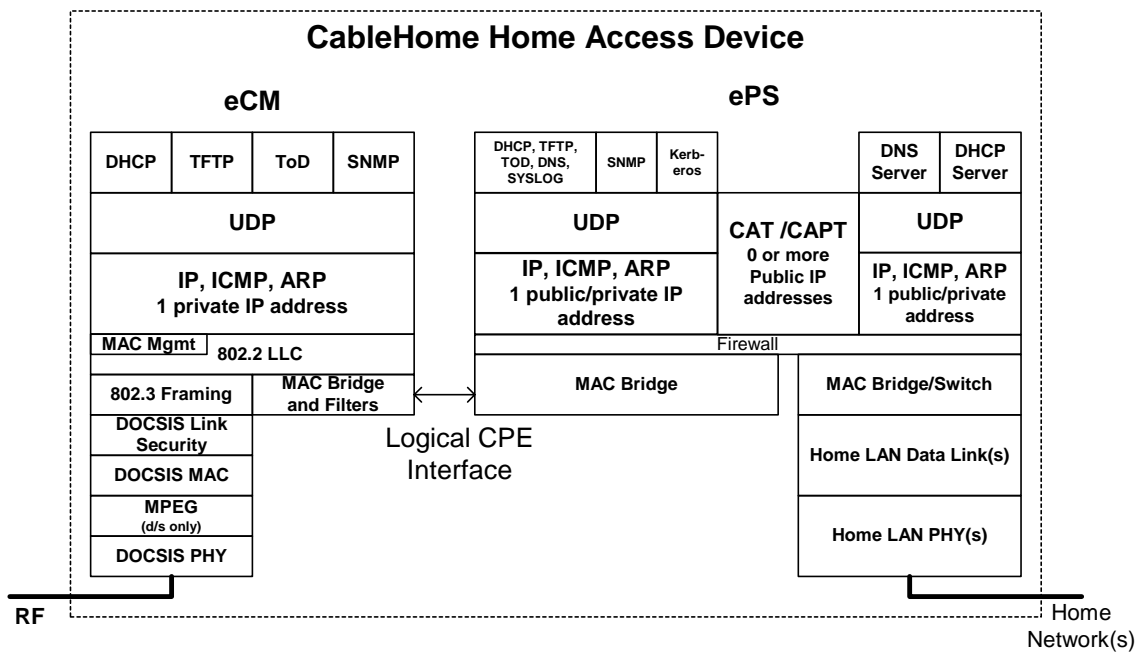


Figure 5-3 - eCM - ePS Protocol Stacks

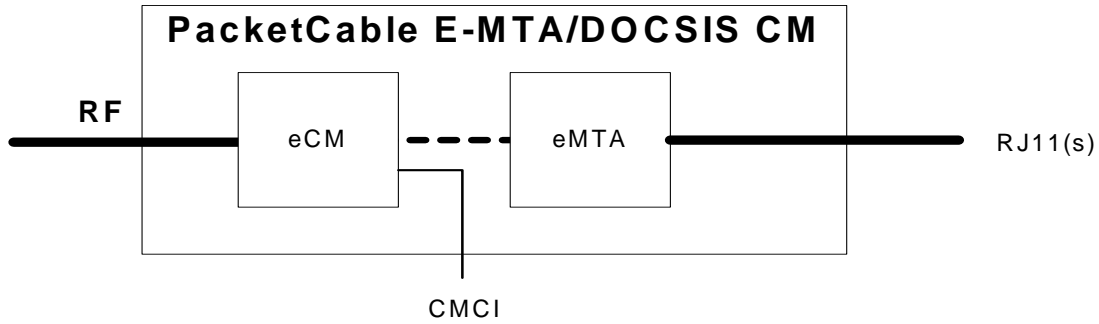


Figure 5-4 - PacketCable E-MTA (with DOCSIS CM) eDOCSIS Reference Model

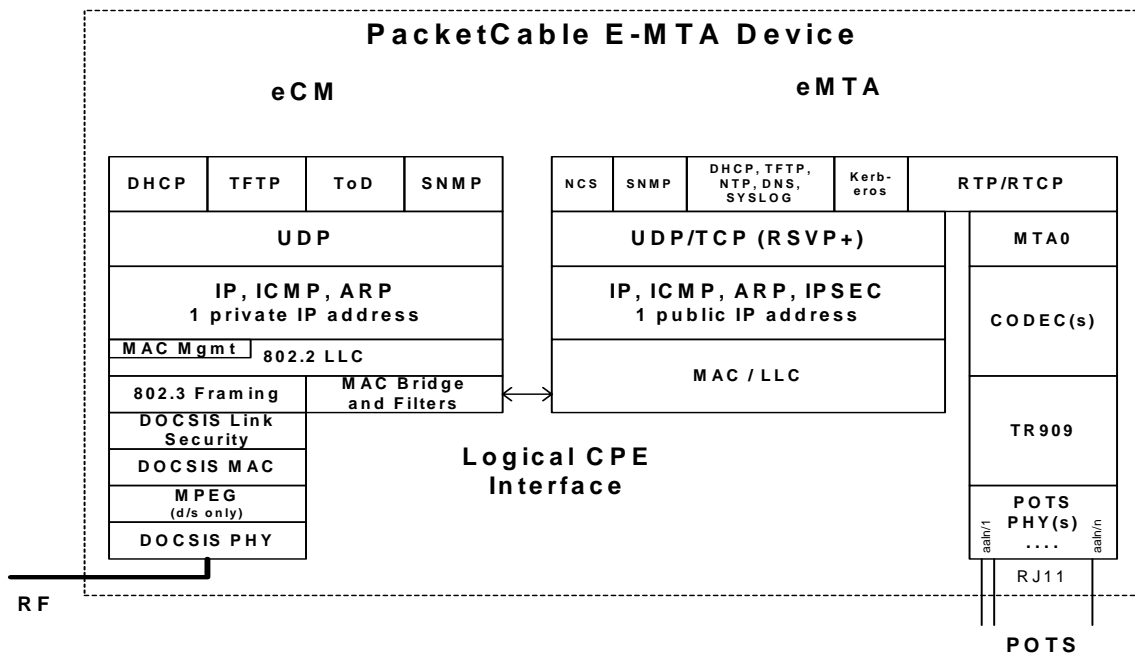


Figure 5-5 - eCM - eMTA Protocol Stacks

5.2 eDOCSIS Requirements

5.2.1 General Requirements

The eCM will provide an SNMP agent which is logically separated from any SNMP agent provided by an eSAFE.

Except for MIB objects that are explicitly allowed to be shared, the DOCSIS-specified MIBs MUST only be accessible through the Management IP address of the eCM. The eSAFE-specified MIBs MUST NOT be accessible through the Management IP address of the eCM.

The MIB objects that MAY be shared are:

- the snmpGroup, systemGroup, udpGroup, icmpGroup and ipGroup (with the exception of ipNetToMediaTable and ipAddrTable within the ipGroup which MUST NOT be shared).
- the mibs rooted under snmpV2 (1.3.6.1.6).

The eCM MUST act as an entity distinct from, and MUST have logical CPE interfaces to, the eSAFEs.

All messages coming from the DOCSIS data network (labeled RF in the diagrams) destined for eSAFEs MUST be processed through the eCM first.

The eCM MUST be the only interface to the DOCSIS data network.

In addition, only a single secured software image download MUST be used for the entire eDOCSIS device and it MUST be controlled by the eCM.

An eCM, MUST meet the requirements of an equivalent standalone cable modem as specified in the applicable DOCSIS Base Specifications. In case any requirement in this document conflicts with a requirement in the DOCSIS Base Specifications (i.e. DOCSIS 1.0, 1.1 and DOCSIS 2.0), the requirement in this document takes precedence for any eDOCSIS Device.⁶

5.2.2 Interface Requirements

The bridging function between RF port and the CPE interfaces (logical or physical) MUST be equivalent to that of a multi-port learning bridge. Each CPE interface MUST comply with the CM Forwarding Rules defined in [DOCSISx-RFI]⁷. In particular:

- MAC addresses seen on the interface are learned or provisioned as defined in [DOCSISx-RFI] specification, and are counted toward the total allowed by the Maximum Number of CPEs configuration setting.
- Packet forwarding and filtering rules defined in [DOCSISx-RFI] specification apply to both logical and physical CPE interfaces as defined in this specification and in [DOCSISx-OSSI]⁸.
- Data forwarding through the interface will obey the Network Access Control Object as defined in [DOCSISx-RFI].

5.2.3 Operations Support Requirements

5.2.3.1 ifTable Requirements

The eCM MUST represent the logical interface to each eSAFE with an entry in the ifTable with ifType other(1) as described in [DOCSISx-OSSI] and as detailed below.

If the eCM is embedded into a device which contains an ePS, then:

- The eCM MUST use ifIndex 1 (the Primary CPE interface) to represent the logical interface between the eCM and the ePS.

⁶ Edited per EDOCS-N-03099, 11/11/03, po

⁷[DOCSISx-RFI] is a shorthand notation for [RFI 1.0, RFI 1.1, and RFI 2.0]

⁸[DOCSISx-OSSI] is a shorthand notation for [OSSI 1.0, OSSI 1.1, and OSSI 2.0]

- The eCM MUST NOT report in the ifTable the physically exposed interfaces associated with the ePS, and MUST NOT report the MIB Module extensions associated with those interfaces (e.g. EtherLike-MIB and USB-MIB).

If the eCM is embedded into a device which contains an eMTA, then:

- The eCM MUST use ifIndex 16 to represent the logical interface between the eCM and the eMTA.
- The eCM MUST NOT report in the ifTable the MTA endpoints (ifType = 198).

The ifXTable MUST be supported in accordance with [RFC 2863]. The Default value of ifLinkUpDownTrapEnable MUST be enabled(1) for logical interfaces to eSAFEs.

The ifStackTable MUST be supported in accordance with [RFC 2863]. The logical interface to an eSAFE MUST NOT contain any sub-layers.

Table 5-1 summarizes the ifIndex assignments in the eCM. Table 5-2 defines the details of the ifTable entries for an ePS or eMTA which MUST be supported.

Table 5-1 - eDOCSIS ifTable Interface Designations

Interface	Type
1	Primary CPE interface (CableHome ePS WAN interface)
2	CATV-MAC
3	RF-downstream channel
4	RF-Upstream channel
5 – 15	Other CPE Interfaces
16	Reserved for PacketCable/eMTA
17 – 31	Reserved for Other eDOCSIS Interfaces

Table 5-2 - [RFC 2863] ifTable, MIB-Object Details for eDOCSIS Device Interfaces

[RFC 2863] MIB-Object details for eCM-eSAFE Interfaces	ePS	eMTA
ifIndex	1	16
ifDescr: MUST match the text	"CableHome Embedded Interface"	"PacketCable Embedded Interface" ⁹
ifType	other(1)	other(1)
ifMtu	0	0
ifSpeed	0	0
ifPhysAddress	<empty-string>	<empty-string> ¹⁰
ifAdminStatus: Only up/own control are required for this interface. Other values are optional	up(1), down(2)	up(1), down(2)
ifOperStatus: Only up/own control are required for this interface. Other values are optional	up(1), down(2)	up(1), down(2)
ifLastChange	<per [RFC 2863]>	<per [RFC 2863]> ¹¹
ifInOctets	(n)	(n)
ifInUCastPkts	(n)	(n)
ifInNUCastPkts	Deprecated	Deprecated
ifInDiscards	0	0
ifInErrors	0	0
ifUnknownProtos	0	0
ifOutOctets	(n)	(n)
ifOutUCastPkts	(n)	(n)
ifOutNUCastPkts	Deprecated	Deprecated
ifOutDiscards	0	0
ifOutErrors	0	0
ifOutQlen	Deprecated	Deprecated
ifSpecific	Deprecated	Deprecated

⁹ Edited per EDOCS-N-03099, 11/11/03, po¹⁰ Edited per EDOCS-N-03099, 11/11/03, po¹¹ Edited per EDOCS-N-03099, 11/11/03, po

5.2.3.2 *ipNetToMediaTable Requirements*

Table 5-3 shows the details of the ipNetToMediaTable entries that MUST be supported by an ePS and eMTA.

Table 5-3 - [RFC 2011] ipNetToMedia MIB-Object Details for eDOCSIS Device Interfaces

[RFC 2011] MIB-Object details for eCM-eSAFE Interfaces	ePS	eMTA
ipNetToMediaIfIndex	1	16
ipNetToMediaPhysAddress	WAN-Man MAC Address	MTA MAC Address
ipNetToMediaNetAddress	WAN-Man Address, if acquired; otherwise 0.0.0.0	MTA Address, if acquired; otherwise 0.0.0.0
ipNetToMediaType	static(4)	static(4)

5.2.3.3 [RFC 1493] Requirements

Ports associated with SAFEs MUST be added to dot1dBasePortTable.

The dot1dTpFdbTable MUST contain information about learned or provisioned CPEs through the logical CPE interfaces with learned(3) or mgmt(5) status accordingly.

All bridge statistics of dot1dTpPortTable MUST be supported for logical CPE interfaces.

5.2.4 DHCP Option 43 Syntax Requirements

In order to facilitate device provisioning, all eDOCSIS devices will use DHCP Option 43 during registration process for providing vendor class identification, embedded component and vendor specific capability enumerations.

5.2.4.1 General Requirements

Each eCM DHCP DISCOVER MUST use Option 43 and its Sub-options 1 through 10 for Vendor Specific Information to identify embedded components as specified in Table 5-4.

Similarly, each eSAFE may issue its own DHCP DISCOVER with Option 43 after eCM has been successfully registered and operational; details are specified in each eSAFE's specification.

5.2.4.2 DHCP Option 43 Syntax

DHCP Option 43 provides device specific information through the use of sub-options. Sub-options 1 through 10 are specified by CableLabs, sub-options 11-127 are reserved for future CableLabs use, and sub-options 128 and above are reserved for vendor use.

The eCM MUST implement the Vendor Specific Information Option (DHCP option 43) as specified in Table 5-4. Details of DHCP option 43 and its sub-options for eDOCSIS are further defined below. The definitions of DHCP Option 43 sub-options MUST conform to requirements imposed by [RFC 2132].

The option begins with a type octet with the value of number 43, followed by a length octet. The length octet is followed by the number of octets of data equal to the value of the length octet. The value of the length octet does not include the two octets specifying the tag and length.

DHCP option 43 in eDOCSIS is a compound option. The content of option 43 is composed of one or more sub-options. Supported DHCP option 43 sub-options in eDOCSIS is in the range 1-254. A sub-option begins with a tag octet containing the sub-option code, followed a length octet which indicates the total number of octets of data. The value of the length octet does not include itself or the tag octet. The length octet is followed by "length" octets of sub-option data.

5.2.4.3 DHCP Option 43 Sub-option Encoding

The encoding of each Option 43 sub-option is defined below. See Table 5-4 for the intended purpose of each sub-option.

The eCM MUST encode DHCP Option 43 sub-option 1 by the number of octets equal to the value of the length octet of this sub-option, with each octet codifying a requested sub-option.

The eCM MUST encode each of the DHCP Option 43 sub-options 2, 3, 4, 5, 6, 7, 8, 9 and 10 as a character string consisting of characters from the NVT ASCII character set, with no terminating NULL.

An eCM MUST send DHCP Option 43 sub-option 2 containing the character string "ECM" (without the quotation marks).

An eCM MUST send DHCP Option 43 sub-option 3 containing a colon-separated list of all eSAFE types in the eDOCSIS device, including at a minimum the colon-separated character string "ECM:<eSAFE>" (without the quotation marks).

Defined eSAFEs are: "EPS" for CableHome embedded Portal Services Element and "EMTA" for PacketCable embedded MTA.

An eCM MUST send DHCP Option 43 sub-option 4 containing the device serial number as in MIB object docsDevSerialNumber.

An eCM MUST send DHCP Option 43 sub-option 5 containing the Hardware version number, identical to the value as reported in <Hardware version> field in MIB object sysDescr.

An eCM MUST send DHCP Option 43 sub-option 6 containing the Software version number, identical to the value as reported in <Software version> field in MIB object sysDescr.

An eCM MUST send DHCP Option 43 sub-option 7 containing the Boot ROM version number, identical to the value as reported in <Boot ROM version> field in MIB object sysDescr.

An eCM MUST send DHCP Option 43 sub-option 8 containing a 3-octet, hexidecimally-encoded, vendor-specific Organization Unique Identifier (OUI) that uniquely identifies the eCM manufacturer. A vendor MAY use the same OUI as in the eCM's MAC address, and MAY use a single OUI to identify all its eDOCSIS products.

An eCM MUST send DHCP Option 43 sub-option 9 containing the Model number, identical to the value as reported in <Model number> field in MIB object sysDescr.

An eCM MUST send DHCP Option 43 sub-option 10 containing the Vendor name, identical to the value as reported in <Vendor name> field in MIB object sysDescr.

In eDOCSIS, DHCP Option 43 sub-option 11-127 are reserved for CableLabs and MUST NOT be used for other purpose.

In eDOCSIS, DHCP Option 43 sub-option 128-254 are reserved for vendors and MAY be used by vendors.

If the total number of octets in all DHCP Option 43 sub-options exceeds 255 octets, the eCM MUST follow [RFC 3396] to split the option into multiple smaller options.

Table 5-4 - DHCP Option 43 Syntax

eCM DHCP Option 43	Value	Description
Sub-option 1		The request sub-option vector is a list of sub-options (within option 43) to be returned to client by the server upon reply to the request. None defined.
Sub-option 2	“ECM”	Device type of the component making the DHCP request. For DOCSIS, this is: <ul style="list-style-type: none"> “ECM”= embedded Cable Modem (as specified by DOCSIS 1.0, 1.1, or 2.0 Base Specifications)
Sub-option 3	“ECM: <eSAFE ₁ >” or “ECM: <eSAFE ₂ >” or “ECM: <eSAFE ₁ >: <eSAFE ₂ >”	Colon-separated list of eCM and eSAFE(s) contained in the complete eDOCSIS device. First on the list MUST be “ECM” for eCM. <eSAFE ₁ > and <eSAFE ₂ > can be “EMTA” or “EPS” corresponding to embedded MTA and embedded Portal Service Element respectively. For example: <ul style="list-style-type: none"> “ECM:EMTA” = A PacketCable/Embedded MTA “ECM:EPS” = A CableHome/Embedded Portal Services Element “ECM:EMTA:EPS” = An Embedded MTA and Embedded Portal Services Element
Sub-option 4	“<device serial number>”	Device serial number as in MIB object docsDevSerialNumber e.g. “123456”
Sub-option 5	“<Hardware version>”	Hardware version number. Identical to value as reported in the <Hardware version> field in MIB object sysDescr. e.g. “v.3.2.1”
Sub-option 6	“<Software version>”	Software version number. Identical to value as reported in the <Software version> field in MIB object sysDescr. e.g. “v.1.0.2”
Sub-option 7	“<Boot ROM version>”	Boot ROM version. Identical to value as reported in the <Boot ROM version> field in MIB object sysDescr. e.g. “Bv4.5.6”
Sub-option 8	“<OUI>”	A 3-octet, hexidecimally-encoded, vendor-specific Organization Unique Identifier (OUI) that may match the OUI in eCM’s MAC address.
Sub-option 9	“<Model number>”	Device model number. Identical to value as reported in the <Model number> field in MIB object sysDescr. e.g. “T3000”
Sub-option 10	“<Vendor name>”	Vendor name or ID. Identical to value as reported in the <Vendor name> field in MIB object sysDescr. e.g. “XYZ Corp”
Sub-options 11 to 127		Reserved for Cablelabs.
Sub-options 128 to 254		Reserved for vendors.

5.2.4.4 Examples (Informative)

5.2.4.4.1 DOCSIS Cable Modem with Embedded PacketCable 1.0 MTA Example

Table 5-5 shows the syntax for DHCP Option 43 for the eCM in an E-MTA.

Table 5-5 - Example eDOCSIS Device: E-MTA

DHCP DISCOVER Options	Value	Description
eCM DHCP DISCOVER		
eCM Option 43 sub-option 1	"<null>"	List of sub-options (within option 43) to be returned by server
eCM Option 43 sub-option 2	"ECM"	Embedded cable modem
eCM Option 43 sub-option 3	"ECM:EMTA"	ECM followed by a list of embedded components (eSAFEs)
eCM Option 43 sub-option 4	"123456"	Device serial number as in MIB object docsDevSerialNumber
eCM Option 43 sub-option 5	"V1.2.3"	Hardware version number as in <Hardware version> field in MIB object sysDescr.
eCM Option 43 sub-option 6	"V3.2.1"	Software version number as in <Software version> field in MIB object sysDescr
eCM Option 43 sub-option 7	"Boot 4.5.6"	Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr
eCM Option 43 sub-option 8	"0204DF"	3-octet OUI as Vendor ID
eCM Option 43 sub-option 9	"Xman200"	Device model number as in <Model number> field in MIB object sysDescr
eCM Option 43 sub-option 10	"XYZ Broadband"	Vendor name as in <Vendor name> field in MIB object sysDescr

5.2.4.4.2 DOCSIS Cable Modem with Embedded CableHome 1.1 PS Example

Table 5-6 shows the syntax for DHCP Options 43 for the eCM in a CableHome 1.1 E-PS device.

Table 5-6 - Example eDOCSIS Device: DOCSIS 1.1 CableHome/E-PS

DHCP DISCOVER Options	Value	Description
eCM DHCP DISCOVER		
eCM Option 43 sub-option 1	"<null>"	List of sub-options (within option 43) to be returned by server
eCM Option 43 sub-option 2	"ECM"	Embedded cable modem
eCM Option 43 sub-option 3	"ECM:EPS"	ECM followed by a list of embedded components (eSAFEs)
eCM Option 43 sub-option 4	"123456"	Device serial number as in MIB object docsDevSerialNumber
eCM Option 43 sub-option 5	"V1.2.3"	Hardware version number as in

		<Hardware version> field in MIB object sysDescr.
eCM Option 43 sub-option 6	"V3.2.1"	Software version number as in <Software version> field in MIB object sysDescr
eCM Option 43 sub-option 7	"Boot 4.5.6"	Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr
eCM Option 43 sub-option 8	"0204DF"	3-octet OUI as Vendor ID
eCM Option 43 sub-option 9	"Xman200"	Device model number as in <Model number> field in MIB object sysDescr
eCM Option 43 sub-option 10	"XYZ Broadband"	Vendor name as in <Vendor name> field in MIB object sysDescr

5.2.4.4.3 DOCSIS Cable Modem with Embedded PacketCable MTA and CableHome PS

Table 5-7 shows the syntax for DHCP Option 43 for the eCM in an eDOCSIS device that contains the following embedded components: a DOCSIS 2.0 eCM, a PacketCable 1.0 eMTA and a CableHome 1.1 ePS.

Table 5-7 - Example eDOCSIS Device Containing DOCSIS 2.0 eCM, eMTA and ePS

DHCP DISCOVER Options	Value	Description
eCM DHCP DISCOVER		
eCM Option 43 sub-option 1	"<null>"	List of sub-options (within option 43) to be returned by server
eCM Option 43 sub-option 2	"ECM"	Embedded cable modem
eCM Option 43 sub-option 3	"ECM:EMTA:EPS"	ECM followed by a list of embedded components (eSAFEs)
eCM Option 43 sub-option 4	"123456"	Device serial number as in MIB object docsDevSerialNumber
eCM Option 43 sub-option 5	"V1.2.3"	Hardware version number as in <Hardware version> field in MIB object sysDescr.
eCM Option 43 sub-option 6	"V3.2.1"	Software version number as in <Software version> field in MIB object sysDescr
eCM Option 43 sub-option 7	"Boot 4.5.6"	Boot ROM version number as in <Boot ROM version> field in MIB object sysDescr
eCM Option 43 sub-option 8	"0204DF"	3-octet OUI as Vendor ID
eCM Option 43 sub-option 9	"Xman200"	Device model number as in <Model number> field in MIB object sysDescr
eCM Option 43 sub-option 10	"XYZ Broadband"	Vendor name as in <Vendor name> field in MIB object sysDescr ¹²

5.2.5 Testability Requirements

In order to verify conformance to this specification and to the DOCSIS Base Specifications, a mechanism to generate and receive traffic bridged through the eCM is required. eDOCSIS devices that have a physically exposed CMCI (e.g. 10BaseT or USB) can be tested by using external packet generation equipment connected to that interface.

For cost, security, or other reasons, however, certain eDOCSIS devices may not have an exposed CMCI, necessitating an alternative mechanism.

Additionally, an eDOCSIS device may have multiple eSAFEs, each with a logical CPE interface (LCI) to the eCM. This specification places requirements on the LCIs as well as the bridging of traffic among eCM and eSAFEs.

To this end, a Software Loopback for eDOCSIS (SLED) is specified below.

¹² Edited per EDOCS-N-03084, 11/11/03, po

5.2.5.1 General Requirements

An eCM SHOULD implement SLED. An eCM without an externally accessible CMCI port, or a physical interface configured to be equivalent to a CMCI port, MUST implement SLED.

5.2.5.2 SLED Protocol Description

5.2.5.2.1 General Description

SLED is an embedded test function residing in an eCM enabling DOCSIS and eDOCSIS conformance testing coverage, particularly when an exposed CMCI is not available.

The SLED test functions are controlled via SLED MIB objects as specified in Annex A. The SLED MIB objects MUST be associated with the SNMP stack of the eCM. The SLED MIB objects MUST NOT be accessible through the CMCI.

To prevent unintended activation of SLED, all SLED functions MUST be disabled by default. SLED functions MUST only be enabled if the MIB object *sledGlobalEnable* is set to 'true' prior to eCM registration; *sledGlobalEnable* may be set to 'true' via inclusion in TLV-11 of the eCM's configuration file.

Figure 5-6 illustrates the SLED reference model.

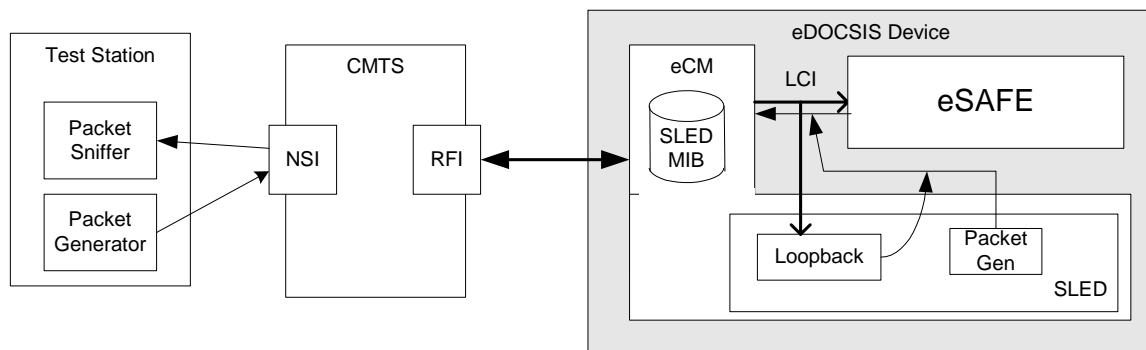


Figure 5-6 - SLED Reference Model

The SLED functionality supports:

1. Packet loopback

The primary purpose of the Packet loopback protocol is to enable verification of the receipt of packets across the LCI by the eSAFE. Once enabled by the SLED MIB object, all packets that are forwarded to the indicated LCI are encapsulated into a pre-defined packet header (Ethernet DIX frame header + IP header + UDP header) and reflected back across the LCI to the eCM for forwarding to the final destination. Typically, the looped-back packets will be addressed to, and captured by, a test station residing in the Network-Side Interface (NSI) of the CMTS.
2. Packet generation

SLED MIB objects are defined to enable setting up of Ethernet framing and payload transmission for packet generation and transmission through the LCI to the eCM. The SLED MIB objects

described below control the packet transmission with parameters such as Ethernet packet header, packet rate, and the number of packets.

3. Packet loopback and packet generation SLED functions MUST be able to be controlled independently.
4. The packet loopback and packet generation SLED functions MUST NOT disrupt network connectivity to or from the eSAFE. When SLED loopback is enabled, a packet transmitted across the LCI in the eCM-to-eSAFE direction, MUST be forwarded to both the eSAFE and the SLED loopback function. When SLED functions are enabled, packets to/from the eSAFE MUST continue to be bridged across the LCI.

5.2.5.2.2 Loopback Protocol

An eCM implementing SLED MUST implement the following loopback protocol:

1. The SLED packet loopback function is attached to the LCI associated with the eSAFE by setting SLED MIB *sledLoopbackInterface* to the eCM's ifIndex number associated with the LCI (per Table 5-1).
2. The SLED MIB object *sledLoopbackPktHdr* is configured with the 42-byte loopback Ethernet packet/IP/UDP headers (14-byte Ethernet header + 20-byte IPv4 header + 8-byte UDP header).
3. As an example, the following loopback header parameters could be used:
 - a) Ethernet MAC source address = eSAFE MAC address
 - b) Ethernet MAC destination address = test station MAC address
 - c) IP source address = eSAFE Management IP address
 - d) IP destination address = test station IP address
 - e) UDP source port number = 7
 - f) UDP destination port number = 7
4. When the SLED MIB object *sledLoopbackInterface* is set to an ifIndex associated with an LCI which supports SLED, *sledLoopbackPktHdr* contains a 42-byte octet string, and *sledLoopbackEnable* is set to 'true', the SLED MUST operate in a loopback mode.
5. When operating in loopback mode, all Ethernet packets forwarded across the indicated LCI by the eCM will be processed as follows:
 - a) If the received Ethernet packet is greater than 1472 octets, the Ethernet packet MUST be split into two fragments according to IP fragmentation scheme as described in [RFC 791], the first consisting of the first 1472 octets of the Ethernet packet and the second containing the remaining octets, resulting in two payloads to that are processed as described below.
 - b) If the received Ethernet packet is less than or equal to 1472 octets, the entire packet MUST be processed as a single payload.
 - c) Each payload generated in step 5a or 5b MUST be prepended with the contents of *sledLoopbackPktHdr*.
 - d) The mutable fields within *sledLoopbackPktHdr* MUST be recomputed. The mutable fields are IP Header Checksum, IP Total Length per [RFC 791], and UDP Checksum, UDP Length per [RFC 768].
 - e) If the Ethernet packet is fragmented as defined in step 5a, the appropriate IP header fields MUST be updated to indicate IP fragmentation. The IP fragmentation header values will differ depending on if this is the first or second fragment being processed (per [RFC 791]). Further, the final 8-bytes of *sledLoopbackPktHdr* (the UDP header) MUST NOT be prepended to the second fragment.
 - f) The Ethernet FCS MUST be computed and appended.

- g) The resulting Ethernet packet MUST be transmitted to the LCI toward the eCM.
- 6. When the SLED MIB object *sledLoopbackEnable* is set to 'false', the SLED loopback function MUST be disabled.
- 7. While the SLED loopback function is enabled, the eCM MUST reject changes to *sledLoopbackInterface* or *sledLoopbackPktHdr*.

Figure 5-7 illustrates the SLED packet loopback encapsulation.

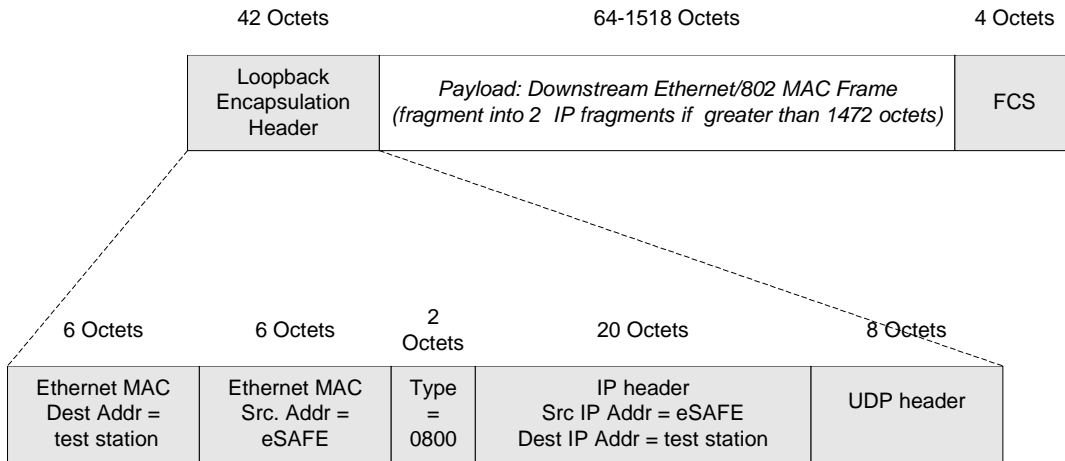


Figure 5-7 - SLED Packet Loopback Encapsulation

Figure 5-8 illustrates an example of the SLED loopback sequence.

5.2.5.2.3 Packet Generation Protocol

An eDOCSIS device implementing SLED MUST implement the following packet generator protocol:

1. The SLED packet generation function is attached to the eCM's LCI associated with the eSAFE by setting SLED MIB *sledPktGenInterface* to the ifIndex number associated with the LCI (per Table 5-1).
2. The SLED MIB object *sledPktGenPayload* is set up to be a complete Ethernet (DIX/802 MAC) packet, including FCS trailer, for transmission across the LCI. The FCS is set to be correct for the packet as specified, and MAY be recalculated by the eCM as required for upstream processing; the SLED is not required to validate the FCS, and a packet with an invalid FCS may be transmitted with a corrected FCS.
3. The SLED MIB objects *sledPktGenRate* and *sledPktGenNumPkts* are set to non-zero values.
4. When *sledPktGenInterface* is set to an ifIndex associated with an LCI which supports SLED, *sledPktGenRate* and *sledPktGenNumPkts* are both set to non-zero values, the SLED Packet Generator MUST start to send generated Ethernet packets to the LCI in within 250 msec after *sledPktGenTrigger* is set to 'start'; the SLED SHOULD start to transmit packets to the LCI as

- soon as possible in order to minimize the amount of time it takes to run tests that use the SLED Packet Generator.
5. When *sledPktGenTrigger* is set to 'start', the SLED Packet Generator MUST set the SLED MIB *sledPktGenLastTrigger* to the current value of the system MIB *sysUptime*.
 6. The packets generated by the SLED Packet Generator MUST be the exact copies of the Ethernet packet specified by the SLED MIB *sledPktGenPayload*. The average rate of generated packets MUST be as specified by the SLED MIB *sledPktGenRate*.
 7. The packet generation MUST be continued until the total number of generated packets reaches the limit as specified by the SLED MIB *sledPktGenNumPkts*, unless terminated by setting *sledPktGenTrigger* to 'stop'. If *sledPktGenTrigger* is set to 'stop' while packets are being generated, the SLED SHOULD stop packet generation within 1 second.
 8. While the previous sequence of SLED packets is still in progress, the eCM MUST reject changes to *sledPktGenInterface*, *sledPktGenPayload*, *sledPktGenNumPkts* or *sledPktGenRate*.

Refer to Figure 5-8 for an illustration of the SLED packet loopback and generation sequences.

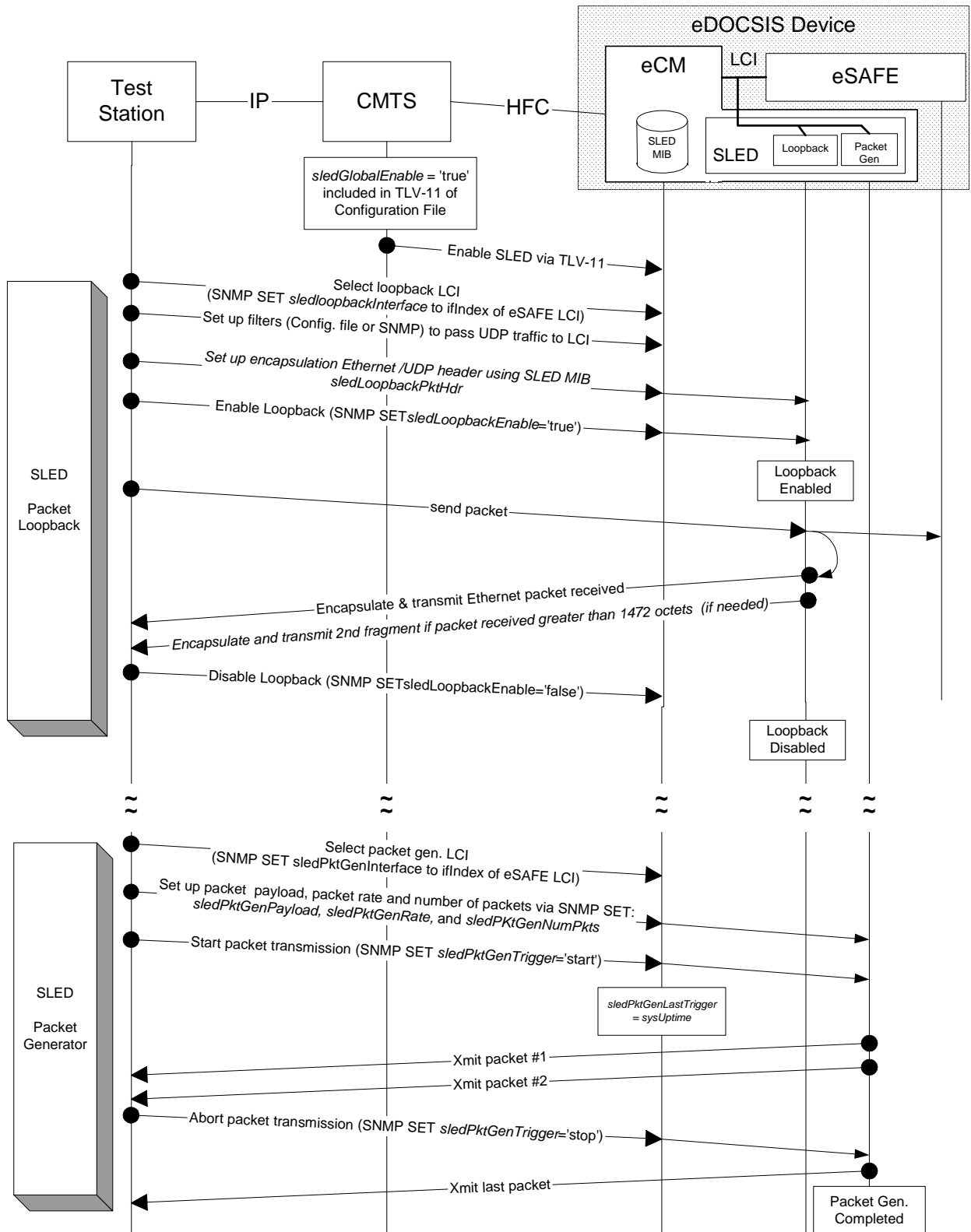


Figure 5-8 - SLED Packet Loopback And Generation Sequences

Annex A SLED MIB Definition

An eCM implementing SLED MUST implement the following SLED MIB objects.

```

SLED-MIB DEFINITIONS ::= BEGIN
IMPORTS
    MODULE-IDENTITY,
        Integer32,
        OBJECT-TYPE          FROM SNMPv2-SMI
        TruthValue,
        TimeStamp            FROM SNMPv2-TC
        OBJECT-GROUP,
        MODULE-COMPLIANCE   FROM SNMPv2-CONF
clabProjDocsis            FROM CLAB-DEF-MIB
InterfaceIndex           FROM IF-MIB
;

sledMib MODULE-IDENTITY
    LAST-UPDATED      "200311170000Z" -- November 17, 2003
    ORGANIZATION      "CableLabs eDOCSIS Working Group"
    CONTACT-INFO
        "John Eng
        Postal:        Cable Television Laboratories, Inc
                    400 Centennial Parkway
                    Louisville, CO 80027
                    U.S.A.
        Phone:        +1 303-661-9100
        Fax:          +1 303-661-9199
        E-mail:       mibs@cablelabs.com"
    DESCRIPTION
        "This MIB module provides the management objects necessary
        to configure and invoke the Software Loopback Application
        for eDOCSIS (SLED) functionality. "
    REVISION           "200311170000Z" -- November 17, 2003
    DESCRIPTION
        "Initial version of the eDOCSIS SLED MIB module.
        This revision is published as part of the CableLabs
        eDOCSIS Specification I02.

        Acknowledgement:
        Greg Nakanishi of Motorola, Inc.

        Copyright 1999-2003 Cable Television Laboratories, Inc.
        All rights reserved."

    ::= { clabProjDocsis 13 }

-- Administrative assignments
sledNotifications      OBJECT IDENTIFIER ::= { sledMib 0 }
sledMibObjects         OBJECT IDENTIFIER ::= { sledMib 1 }
sledMibNotificationsObjects OBJECT IDENTIFIER ::= { sledMib 2 }
sledMibConformance    OBJECT IDENTIFIER ::= { sledMib 3 }

-- Object Groups
sledGlobal             OBJECT IDENTIFIER ::= { sledMibObjects 1 }
sledLoopback          OBJECT IDENTIFIER ::= { sledMibObjects 2 }
sledPktGen             OBJECT IDENTIFIER ::= { sledMibObjects 3 }

--

```

```
-- The following group describes the objects that apply to
-- both loopback and packet generator SLED functionality
--
sledGlobalEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "This object allows the SLED functionality to be
        enabled/disabled.
        This object may only be updated prior to device registration.
        If the device has completed registration, any attempt to
        update the value of this object returns 'notWritable'.
        Prior to registration, if the value of this object is set to
        'true', the SLED functionality is enabled and access to this
        MIB is allowed.
        Prior to registration, if the value of this object is set to
        'false', the SLED functionality is disabled and any attempt to
        update other objects in this MIB returns 'noAccess'."
    DEFVAL {false}
    ::= { sledGlobal 1 }

--
-- The following group describes the loopback objects
--
sledLoopbackInterface OBJECT-TYPE
    SYNTAX      InterfaceIndex
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The index of the logical CPE interface (LCI) that the SLED
        loopback function is attached to. If the index does not
        correspond to a LCI supported by this device, 'wrongValue'
        is returned.
        Any attempt to set this object while sledLoopbackEnable is
        set to 'true' returns 'notWritable'."
    ::= { sledLoopback 1 }

sledLoopbackEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Setting this object to 'true' enables the loopback function.
        Setting this object to 'false' disables the loopback function.
        When enabled, all Ethernet packets received by the SLED from the
        LCI are processed as follows:

        1. If the received Ethernet packet is greater than 1472 octets,
           the Ethernet packet is split into two fragments, the first
           consisting of the first 1472 octets of the Ethernet packet
           and the second containing the remaining octets, resulting
           in two payloads that are processed as described below. If
           the received Ethernet packet is less than or equal to 1472
           octets, the entire packet will be processed as a single
           payload.
        2. For each payload generated in step 1, the payload is
           appended to the contents of sledLoopbackPktHdr.
        3. The mutable fields within sledLoopbackPktHdr MUST be
           recomputed. The mutable fields are IP Header Checksum, IP
```

- Total Length, UDP Checksum, and UDP Length.
4. If the Ethernet packet was fragmented in step 1, the appropriate IP header fields (Flags and Fragment Offset) are updated to indicate IP fragmentation. These IP fragmentation header values will differ depending on if this is the first or second fragment being processed.
 5. The Ethernet FCS is computed and appended.
 6. The resulting Ethernet packet is transmitted to the LCI."

```
DEFVAL {false}
 ::= { sledLoopback 2 }
```

sledLoopbackPktHdr OBJECT-TYPE

```
SYNTAX      OCTET STRING (SIZE(42))
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"A properly formatted Ethernet(DIX)+IP+UDP headers for use in SLED loopback processing as described in sledLoopbackEnable. The object value contains mutable fields that are recomputed: the IP Header Checksum, IP Total Length, UDP Length, and UDP Checksum. Any attempt to set this object while sledLoopbackEnable is set to 'true' returns 'notWritable'."

```
 ::= { sledLoopback 3 }
```

--

-- The following group describes the packet generation objects

--

sledPktGenInterface OBJECT-TYPE

```
SYNTAX      InterfaceIndex
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"The index of the logical CPE interface (LCI) that the SLED packet generation function is attached to. If the index does not correspond to a LCI supported by the device, 'wrongValue' is returned. Any attempt to set this object while sledPktGenTrigger is set to 'start' returns 'notWritable'."

```
 ::= { sledPktGen 1 }
```

sledPktGenPayload OBJECT-TYPE

```
SYNTAX      OCTET STRING (SIZE(64..1518))
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"The properly formatted Ethernet packet payload to be generated. Any attempt to set this object while sledPktGenTrigger is set to 'start' returns 'notWritable'."

```
 ::= { sledPktGen 2 }
```

sledPktGenRate OBJECT-TYPE

```
SYNTAX      Integer32
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
```

"The packet rate (in packets per second) that the SLED is to transmit the packet specified in the sledPktGenPayload. Any attempt to set this object while sledPktGenTrigger is set to

```

        'start' returns 'notWritable'."
        DEFVAL {10}
        ::= { sledPktGen 3 }

sledPktGenNumPkts OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Number of packets to be generated at the rate specified by
        sledPktGenRate. Any attempt to set this object while
        sledPktGenTrigger has been set to 'start' will return
        'notWritable'."
    DEFVAL {1}
    ::= { sledPktGen 4 }

sledPktGenTrigger OBJECT-TYPE
    SYNTAX      INTEGER {
        start(1),
        stop(2)
    }
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "This object controls the packet generation. Setting this object
        to 'start' causes the packet generation to begin. Reading this
        object will return 'start' if a packet generation is in progress,
        otherwise it will return 'stop'. Setting this object to 'stop'
        while packet generation is in progress aborts the packet
        generation. Setting this object to 'start' while packet
        generation is in progress, 'wrongValue' is returned."
    DEFVAL {stop}
    ::= { sledPktGen 5 }

sledPktGenLastTrigger OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Value of sysUptime when the packet generation was
        last triggered."
    ::= { sledPktGen 6 }

-- Conformance information *****

sledMibCompliances OBJECT IDENTIFIER ::= { sledMibConformance 1 }
sledMibGroups      OBJECT IDENTIFIER ::= { sledMibConformance 2 }

-- Compliance statements

sledMibCompliance MODULE-COMPLIANCE
    STATUS      current
    DESCRIPTION
        "The compliance statement for SLED."
    MODULE

-- unconditionally mandatory groups

    MANDATORY-GROUPS {
        sledMibBaseGroup

```

```
    }  
 ::= { sledMibCompliances 1 }  
  
sledMibBaseGroup OBJECT-GROUP  
  OBJECTS {  
    sledGlobalEnable,  
    sledLoopbackInterface,  
    sledLoopbackEnable,  
    sledLoopbackPktHdr,  
    sledPktGenInterface,  
    sledPktGenPayload,  
    sledPktGenRate,  
    sledPktGenNumPkts,  
    sledPktGenTrigger,  
    sledPktGenLastTrigger  
  }  
  STATUS      current  
  DESCRIPTION  
    "Group of object in SLED MIB"  
 ::= { sledMibGroups 1 }
```

END
13

¹³ ECOCS-N-03088 11/17/03 added new figure 5-1 and section 5.2.5 thru Annex A

Appendix A Acknowledgements

The inter-disciplinary nature of this specification involves contributions from many individuals. On behalf of CableLabs and its participating member companies, I would like to extend our sincere appreciation to all those have contributed to the development of this specification. Special thanks are given to:

Ralph Brown CableLabs
Eduardo Cardona..... CableLabs
Kevin Leuhrs..... CableLabs
Matt Osman..... CableLabs
Joe Weber..... CableLabs
Greg White..... CableLabs
Margo Dolas..... Broadcom
Gordon Li..... Conexant
Sridhar Sharma..... Imedia
Greg Nakanishi Motorola
Mariano Schain Texas Instruments
Nancy Davoust..... YAS
Doug Jones..... YAS

Special thanks are also given to Liz Weeks of YAS for her skillful editing work making the aggressive release deadlines possible.

John Eng, CableLabs.

Appendix B Revision History

The following Engineering Change Notices have been incorporated in SP-e-DOCSIS I02-031117.

ECN	Date Ratified	Summary
EDOCS-N-03084	8/20/03	Incorporate DHCP Option 43 for eDOCSIS.
EDOCS-N-03099	10/29/03	Corrections and Clarifications; alignment of rearranged text with J.eCM submission.
EDOCS-N-03088	10/29/03	Incorporate Software Loopback for eDOCSIS (SLED).