

International Telecommunication Union

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**J.214**

(07/2007)

SERIES J: CABLE NETWORKS AND TRANSMISSION  
OF TELEVISION, SOUND PROGRAMME AND OTHER  
MULTIMEDIA SIGNALS

Interactive systems for digital television distribution

---

## **Cable modem TDM emulation interface**

ITU-T Recommendation J.214





# **ITU-T Recommendation J.214**

## **Cable modem TDM emulation interface**

### **Summary**

TDM emulation service (TDM-E) is a method for cable operators to deliver T1, E1 and NxDS0 emulation services that meet or exceed the quality requirement of applications that use such services. ITU-T Recommendation J.214 is part of the CableModem family of Recommendations and, in particular, defines the TDM-E architecture and components.

### **Source**

ITU-T Recommendation J.214 was approved on 29 July 2007 by ITU-T Study Group 9 (2005-2008) under the ITU-T Recommendation A.8 procedure.

## FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

## NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

## INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2008

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

## CONTENTS

	<b>Page</b>	
1	Scope and purpose .....	1
1.1	Scope .....	1
1.2	Introduction and overview .....	1
2	References.....	2
2.1	Normative references.....	2
2.2	Informative references.....	4
2.3	Reference acquisition .....	4
3	Terms and definitions .....	5
3.1	Terms defined elsewhere.....	5
3.2	Terms defined in this Recommendation.....	5
4	Abbreviations, acronyms and conventions .....	6
4.1	Abbreviations and acronyms .....	6
4.2	Conventions.....	7
5	Technical overview.....	8
5.1	Background.....	8
5.2	Principal technical considerations .....	8
5.3	Devices/Interfaces .....	9
5.4	Architecture .....	13
5.5	Network environment.....	16
6	Technical specification .....	20
6.1	General requirements.....	20
6.2	TDM-IP network interworking.....	23
6.3	Payload formats .....	26
6.4	Timing aspects.....	30
6.5	Defects and monitoring .....	43
6.6	Security.....	44
6.7	Provisioning and service initiation .....	44
6.8	Management .....	64
Annex A	– Event, SYSLOG, and SNMP trap extensions .....	70
A.1	TEA events description .....	70
A.2	DOCSIS events extensions.....	70
Annex B	– TEA MIB definition .....	73
B.1	CL-PW-TC-MIB .....	73
B.2	CL-PW-MIB.....	77
B.3	CL-PW-TDM-MIB.....	109

	<b>Page</b>
Appendix I – Operator recommendations .....	143
I.1 Preferential treatment for TE-CMs .....	143
I.2 Service level agreement considerations .....	143
I.3 Standard DOCSIS operations .....	144
Appendix II – Delay aspects .....	146
II.1 Upstream delay .....	146
II.2 Downstream delay .....	147
Appendix III – Bandwidth usage .....	149
III.1 Upstream bandwidth usage.....	149
III.2 Downstream bandwidth usage.....	150
Appendix IV – eTEA configuration example .....	151
IV.1 Structure agnostic T1 configuration .....	151
IV.2 Structure agnostic E1 configuration .....	152
IV.3 Structure aware 5xDS0 configuration .....	153
IV.4 Structure aware 5xDS0 configuration with CAS .....	154
IV.5 Structure aware T1 with RTP configuration .....	155
Appendix V – Clocking options, by example .....	157
V.1 Background.....	157
V.2 Clocking modes .....	158
V.3 Clocking decision tree .....	161
Bibliography.....	162

# ITU-T Recommendation J.214

## Cable modem TDM emulation interface

### 1 Scope and purpose

#### 1.1 Scope

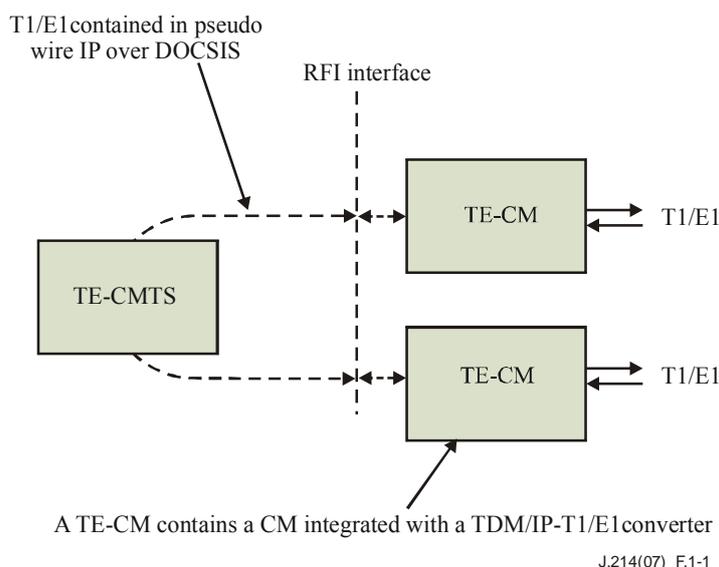
TDM Emulation service (TDM-E) is a method for cable operators to deliver T1, E1 and NxDS0 emulation services that meet or exceed the quality requirement of applications that use such services. This Recommendation is part of the CableModem family of Recommendations and, in particular, defines the TDM-E architecture and components.

NOTE – The structure and content of this Recommendation have been organized for ease of use by those familiar with the original source material; as such, the usual style of ITU-T Recommendations has not been applied.

#### 1.2 Introduction and overview

In legacy telecommunication networks, telephone calls are often brought into households, one at a time, over twisted pair wires. To transport many telephone calls at once (i.e., between business, wireless base stations and in the telephone network), single calls are time-multiplexed together into 'T1' signals. A single T1 signal carries 24 individual calls, and a similar 'E1' signal carries 32 calls. Since T1 and E1 (T1/E1) services have been deployed for quite some time, the performance standards, tariffs and market are well defined. A number of ITU, ETSI and ANSI standards define the various aspects of T1/E1 services. Moreover, the usage and deployment models of T1/E1 lines are well understood.

This Recommendation outlines the methods by which T1/E1 structured, unstructured, and fractional signals can be converted to IP packets, transported over a DOCSIS IP network, and converted back to T1/E1 signals with high reliability and quality. Figure 1-1 shows a simple T1 service delivered between two customer sites over DOCSIS.



**Figure 1-1 – Example of T1 service delivery**

## 2 References

### 2.1 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revisions; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [ITU-T G.703] ITU-T Recommendation G.703 (2001), *Physical/electrical characteristics of hierarchical digital interfaces*.
- [ITU-T G.704] ITU-T Recommendation G.704 (1998), *Synchronous frame structures used at 1544, 6312, 2048, 8448 and 44 736 kbit/s hierarchical levels*.
- [ITU-T G.706] ITU-T Recommendation G.706 (1991), *Frame alignment and cyclic redundancy check (CRC) procedures relating to basic frame structures defined in Recommendation G.704*.
- [ITU-T G.732] ITU-T Recommendation G.732 (1988), *Characteristics of primary PCM multiplex equipment operating at 2048 kbit/s*.
- [ITU-T G.810] ITU-T Recommendation G.810 (1996), *Definitions and terminology for synchronization networks*.
- [ITU-T G.823] ITU-T Recommendation G.823 (2000), *The control of jitter and wander within digital networks which are based on the 2048 kbit/s hierarchy*.
- [ITU-T G.824] ITU-T Recommendation G.824 (2000), *The control of jitter and wander within digital networks which are based on the 1544 kbit/s hierarchy*.
- [ITU-T G.1020] ITU-T Recommendation G.1020 (2006), *Performance parameter definitions for quality of speech and other voiceband applications utilizing IP networks*.
- [ITU-T G.1050] ITU-T Recommendation G.1050 (2007), *Network model for evaluating multimedia transmission performance over Internet protocol*.
- [ITU-T J.83] ITU-T Recommendation J.83 (2007), *Digital multi-program systems for television, sound and data services for cable distribution*.
- [ITU-T J.112] ITU-T Recommendation J.112 (1998), *Transmission systems for interactive cable television services*.
- [ITU-T J.122] ITU-T Recommendation J.122 (2007), *Second-generation transmission systems for interactive cable television services – IP cable modems*.
- [ITU-T J.126] ITU-T Recommendation J.126 (2007), *Embedded cable modem device specification*.
- [ITU-T J.210] ITU-T Recommendation J.210 (2006), *Downstream RF interface for cable modem termination systems*.

- [ITU-T J.211] ITU-T Recommendation J.211 (2006), *Timing interface for cable modem termination systems.*
- [ITU-T J.212] ITU-T Recommendation J.212 (2006), *Downstream external physical layer interface for modular cable modem termination systems.*
- [ITU-T J.222.1] ITU-T Recommendation J.222.1 (2007), *Third-generation transmission systems for interactive cable television services – IP cable modems: Physical layer specification.*
- [ITU-T J.222.2] ITU-T Recommendation J.222.2 (2007), *Third-generation transmission systems for interactive cable television services – IP cable modems: MAC and upper layer protocols .*
- [ITU-T X.690] ITU-T Recommendation X-690 (2002) |ISO/IEC 8825-1:2002, *Information technology – ASN.1 encoding rules: Specification of Basic Encoding rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).*
- [DOCSIS OSSv2.0] ANSI/SCTE 79-2 (2002), *DOCS 2.0 Operations Support System Interface.*
- [IETF RFC 1123] IETF RFC 1123 (1989), *Requirements for Internet Hosts – Application and Support.*
- [IETF RFC 1157] IETF RFC 1157 (1990), *A Simple Network Management Protocol (SNMP).*
- [IETF RFC 1350] IETF RFC 1350 (1992), *The TFTP Protocol (Revision 2).*
- [IETF RFC 2131] IETF RFC 2131 (1997), *Dynamic Host Configuration Protocol.*
- [IETF RFC 2132] IETF RFC 2132 (1997), *DHCP Options and BOOTP Vendor Extensions.*
- [IETF RFC 2494] IETF RFC 2494 (1999), *Definitions of Managed Objects for the DS0 and DS0 Bundle Interface Type.*
- [IETF RFC 2495] IETF RFC 2495 (1999), *Definitions of Managed Objects for the DS1, E1, DS2 and E2 Interface Types.*
- [IETF RFC 2670] IETF RFC 2670 (1999), *Radio Frequency (RF) Interface Management Information Base for MCNS/DOCSIS compliant RF interfaces.*
- [IETF RFC 2863] IETF RFC 2863 (2000), *The Interfaces Group MIB.*
- [IETF RFC 3396] IETF RFC 3396 (2002), *Encoding Long Options in the Dynamic Host Configuration Protocol (DHCPv4).*
- [IETF RFC 3550] IETF RFC 3550 (2003), *RTP: A Transport Protocol for Real-Time Applications.*
- [IETF RFC 3611] IETF RFC 3611 (2003), *RTP Control Protocol Extended Reports (RTCP XR).*
- [IETF RFC 3635] IETF RFC 3635 (2003), *Definitions of Managed Objects for the Ethernet-like Interface Types.*
- [IETF RFC 3636] IETF RFC 3636 (2003), *Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs).*
- [IETF RFC 3711] IETF RFC 3711 (2004), *The Secure Real-time Transport Protocol (SRTP).*

- [IETF RFC 3895] IETF RFC 3895 (2004), *Definitions of Managed Objects for the DS1, E1, DS2, and E2 Interface Types*.
- [IEEE 1588] IEEE 1588 (2002), *Standard for a Precision Clock Synchronization Protocol for networked measurement and Control Systems*. Also *IEC 61588 (2004)*.

## 2.2 Informative references

This Recommendation uses the following informative references.

- [ANSI T1.403] ANSI T1.403-1999, *Network and Customer Installation Interfaces (ISDN) primary rate – Layer 1– Electrical Interfaces Specification*.
- [CMCI] *Data-Over-Cable Service Interface Specifications Cable Modem to Customer Premise Equipment Interface Specification*, SP-CMCI-110-050408, April 8, 2005, Cable Television Laboratories, Inc.
- [IANA UDP PORTS] Internet Assigned Numbers Authority, Port Numbers, April 2006, <http://www.iana.org/assignments/port-numbers>
- [ID PW MIB] T. Nadeau, D. Zelig, *Pseudowire (PW) Management Information Base (MIB)*, draft-ietf-pwe3-pw-mib-14, July 2008.
- [ID PWTC] T. Nadeau, D. Zelig, *Definitions of Textual Conventions for Pseudowires (PW) Management*, draft-ietf-pwe3-pw-tc-mib-14, July 2008.
- [ID VCCV] T. Nadeau, C. Pignataro, *Pseudowire Virtual Circuit Connectivity Verification (VCCV), A Control Channel for Pseudowires*, draft-ietf-pwe3-vccv-15, March 2008.
- [IETF RFC 3551] IETF RFC 3551 (2003), *RTP Profile for Audio and Video Conferences with Minimal Control*.
- [IETF RFC 4553] IETF RFC 4553 (2006), *Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)*.
- [IETF RFC 4928] IETF RFC 4928 (2007), *Avoiding Equal Cost Multipath Treatment in MPLS Networks*.
- [MLDA] Sisalem D. and A. Wolisz, *MLDA: A TCP-friendly Congestion Control Framework for Heterogeneous Multicast Environments*, *IEEE Proc. IWQoS 2000*.
- [TIA NIM] TR41.4-03-05-029 (2003), *IP Network Impairment Simulation for VoIP Performance Testing*.
- [TR-62411] AT&T Technical Reference 62411, *AT&T ACCUNET T1.5 Service Description and Interface Specification*, December 1990.

## 2.3 Reference acquisition

- American National Standards Institute, Internet: <http://webstore.ansi.org>
- AT&T, <http://www.att.com>
- Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027; Phone 303-661-9100; Fax 303-661-9199; Internet: <http://www.cablelabs.com>
- Institute of Electrical and Electronics Engineers (IEEE), <http://www.ieee.org>
- Internet Engineering Task Force (IETF), <http://www.ietf.org>

- International Telecommunication Union – Telecommunication Standardization Sector (ITU-T), <http://www.itu.int/itu-t/>
- International Organization for Standardization (ISO), <http://www.iso.org>
- Metro Ethernet Forum (MEF), <http://www.metroethernetforum.org>
- Society of Cable Telecommunications Engineers (SCTE), <http://www.scte.org>
- Telecommunications Industry Association (TIA), <http://www.tiaonline.org/>
- Telcordia Technologies, <http://www.telcordia.com/>

### 3 Terms and definitions

#### 3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

**3.1.1 jitter:** The fluctuation in the arrival time of a regularly scheduled event such as a clock edge or a packet in a stream of packets. Jitter is defined in [ITU-T G.810] as fluctuations above 10 Hz. See wander.

**3.1.2 maximum time interval error (MTIE):** [ITU-T G.810].

**3.1.3 time deviation (TDEV):** [ITU-T G.810].

**3.1.4 wander:** Low speed jitter. Wander is defined in [ITU-T G.810] as fluctuations below 10 Hz. See jitter.

#### 3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

**3.2.1 alarm indication signal:** Also known as the *blue alarm*. When no incoming signal is detected, a CSU/DSU transmits an unframed all-ones pattern to the network to maintain synchronization and announce its presence to the network.

**3.2.2 clock unit (CU):** A clock unit performs translations and distribution of TDM clocking information across (and between) physical, data and network layers.

**3.2.3 clock recovery unit (CRU):** A clock recovery unit exists in the IWF and is responsible for regenerating the circuits clock based on the average inter-arrival time of the packets in the adaptive clocking mode, or on the time stamp differentials received when operating in the differential clock mode. The output clock is provided to the CU.

**3.2.4 channel service unit:** The piece of a CSU/DSU that talks to the telco network, understands framing and line coding, and provides electrical isolation of the network from the telco network.

**3.2.5 data service unit:** The part of the CSU/DSU that interfaces with routers, switches and packets. It has a serial port to interface with compatible data equipment.

**3.2.6 digital signal 0 (DS0):** Digital signal 0 (DS0) is a basic digital signalling rate of 64 kbit/s, corresponding to the capacity of one voice-frequency-equivalent channel.

**3.2.7 E1:** E1 is a physical layer telephony protocol carrying data at 2048 Mbit/s. It can carry up to 32 DS0s, each of which can carry a telephone conversation.

**3.2.8 eTEA:** An eTEA is an eDOCSIS eSAFE; an embedded version of a TEA.

**3.2.9 fractional:** A fractional T1 or fractional E1 carries only a portion of the total number of DS0s that a T1 or E1 carries.

**3.2.10 interworking function (IWF):** An interworking function (IWF) is a logical entity. It consists of a TSP data interface on one side, an IP packet interface on the other side, and the functionalities to encode TDM data into a pseudo wire in one direction and decode TDM data from a pseudo wire in the other direction.

**3.2.11 logical CPE interface (LCI):** A logical CPE interface (LCI) is a logical 802.3/Ethernet MAC frame data interface.

**3.2.12 loss of frame:** Also called the *red alarm*. When a T1 CSU/DSU is unable to synchronize framing patterns with the remote end for 2.5 seconds, LoF is declared.

**3.2.13 loss of signal:** When no incoming pulses are received by a T1 CSU/DSU for a prescribed number of bit times, LoS is declared. Even if only zeros were transmitted as data, some framing bits should result in pulses on the line during that time. If LoS persists, LoF will eventually be declared because there is no incoming signal with which to synchronize.

**3.2.14 out of frame:** When frame synchronization is lost, an OoF event is recorded. If OoF persists, LoF is declared. OoF is cleared when frame synchronization is regained.

**3.2.15 pseudo wire:** A pseudo wire (PW) is a logical entity capable of the emulation of a native service (i.e., T1) over a packet switched network (like DOCSIS IP).

**3.2.16 remote alarm indication:** Also known as *yellow alarm*. When a CSU/DSU enters the red alarm state, an RAI is transmitted in the outgoing direction. The RAI signals to the remote end that the local end is unable to synchronize framing patterns.

**3.2.17 T1:** T1 is a physical layer telephony protocol carrying data at 1544 Mbit/s. It can carry up to 24 DS0s, each of which can carry a telephone conversation. It is used mainly in North America.

**3.2.18 TDM interface:** A physical time domain multiplex (TDM) telephony interface such as T1 or E1. Time division multiplexing (TDM) is the means by which multiple digital signals (DS0s) can be carried on a single transmission path by interleaving portions of each signal in time.

**3.2.19 TDM emulation adapter (TEA):** A logical entity containing various functions to provide a TDM-E circuit emulation service.

**3.2.20 TDM emulation cable modem (TE-CM):** A special purpose cable modem that integrates a DOCSIS eCM with an eTEA.

**3.2.21 TE-CMCTS:** A TE-CMCTS is a CMCTS or M-CMCTS that can support a TE-CM and can meet the minimum clocking and performance goals to support TDM-E traffic. A TE-CMCTS may or may not support integrated TEAs.

**3.2.22 TDM service processor (TSP):** Consists of a TSP data interface on one side, a TDM interface on the other side, and the functionalities to encapsulate TSP data into TDM interface signals in one direction and extract TSP data from TDM interface signals in the other direction.

## 4 Abbreviations, acronyms and conventions

### 4.1 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ACR	Adaptive Clock Recovery
AIS	Alarm Indication Signal
A-TDMA	Advanced Time Division Multiple Access
CBR	Constant Bit Rate
CM	Cable Modem

CMTS	Cable Modem Termination System
CSU	Channel Service Unit
DEPI	DOCSIS External PHY Interface
DOCSIS®	Data-Over-Cable Service Interface Specification
DS	Downstream
DSU	Data Service Unit
DTI	DOCSIS Timing Interface
HFC	Hybrid Fibre Coax
IP	Internet Protocol
LoF	Loss of Frame
LoS	Loss of Signal
MAC	Media Access Control
OoF	Out of Frame
PDCR	Prime Differential Clock Recovery
PW	Pseudo Wire
RAI	Remote Alarm Indication
RF	Radio Frequency
UDP	User Datagram Protocol
UGS	Unsolicited Grant Service

## 4.2 Conventions

Throughout this Recommendation, 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 Recommendation.
"MUST NOT"	This phrase means that the item is an absolute prohibition of this Recommendation.
"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 behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour 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, i.e., another vendor may omit the same item.

## 5 Technical overview

### 5.1 Background

Throughout the evolution of communication networks, the T1 and E1 service that is delivered to the customer has remained relatively consistent (in large part due to the strict ITU and ANSI standards); however, the method of delivering the T1 or E1 service has changed to increase efficiencies and leverage new technologies. Both SONET/SDH and ATM historically have been adapted to transport T1 and E1 services; now MPLS and IP networks are also being used. Various standards bodies and vendors have created methods for circuit emulation of T1 and E1 services across IP networks, leveraging the successful ATM circuit emulation technology developed in the 1990s. This Recommendation leverages the work done in the ITU and IETF to define circuit emulation of T1 and E1 over a DOCSIS network.

This Recommendation describes the architecture and components of a network that delivers emulated T1 or E1 or NxDS0 services over DOCSIS. It defines the T1, E1, or NxDS0 service delivered to the end customer, the requirements of a TDM-E compliant cable modem (TE-CM), the requirements of a TDM-E compliant CMTS (TE-CMTS), the requirements of the TDM emulation adapter (TEA), and pseudo wires (PW) that cross the packet network connecting two TEAs.

In developing this Recommendation, the following assumptions were made concerning the implementation and deployment of TDM-E systems:

- Interoperation with DOCSIS CMTS versions and options:  
Existing CMTSs can lock their existing DOCSIS symbol clock or DOCSIS SYNC message generation to master clock inputs for both A-TDMA and S-CDMA. The chain of clock synchronization is that Stratum 1 Traceable Reference drives the DOCSIS master clock; the master clock drives the DOCSIS downstream timing; and the CM recovers timing from the DOCSIS downstream.  
Some CM silicon originally developed for DOCSIS 1.1 or 2.0 residential data modems can be used in a design where the symbol clock or DOCSIS SYNC message may be recovered to generate a T1/E1 clock.
- TE-CMs are always connected to a TE-CMTS with QoS support for constant bit rate flows. The TE-CMTS complies with DOCSIS 1.1 or 2.0. A DOCSIS 1.0 CMTS will not be able to host TE-CMs. DOCSIS 1.1 may be limited in upstream bandwidth; TE-CM data flows are symmetric upstream and downstream.
- It is not in the scope of this Recommendation to cover interoperation between dissimilar services, i.e., between E1 and T1.
- The following requirements are market specific and are out of scope of this Recommendation:
  - Path redundancy and equipment redundancy (like redundant power supplies);
  - Environmental hardening, such as clock stability over temperature.
- There is not sufficient need to support the 'octet-aligned format' of [IETF RFC 4553] section 5.2, which is designed to simplify the data handling in and out of a Sonet/SDH virtual tributary/container. This Recommendation does not support that format.

### 5.2 Principal technical considerations

This clause describes the principle technologies and operational considerations germane to the development of a T1/E1-bearing TE-CM and its use. Major sub-topics include:

- Clocking – T1/E1 transport technology is synchronous in nature; requiring accurate and stable clocking. Since T1 transport requires that T1 signals are handed from one service provider to another, each service provider must adhere to the same clocking requirements.

Further, these synchronous clocking requirements must be very stringent to ensure reliable operation. The clause on clocking (clause 5.5) will further describe these requirements largely based on existing ITU and ATIS requirements.

- Data protocol
  - 1) Voice protocol – Phone calls at termination are completely analog, carrying linear voice signals over dedicated wires. It was not long before the phone companies realized that a wire could carry more than one phone call if the calls were time-multiplexed on the wire. Calls are now time-sampled, and the resulting digital signals are time-multiplexed over wires in T1/E1 formats (among others).
  - 2) Data protocol – It did not take long for engineers to realize that T1/E1 transport could be used for standard data (not just voice). T1 signals are now used to transport digital data in many different data formats which are beyond the scope of this Recommendation. This Recommendation accommodates the transport of either voice or data by specifying more than one data format for the IP packets that carry the T1 information over DOCSIS channels. The clauses below addressing 'structure agnostic' protocol are germane to data transport. Most of this Recommendation addresses T1 transport of voice calls, but the Recommendation accommodates data transport as well.
  - 3) Signalling data protocol – To successfully combine several voice phone calls into a T1 structure, data describing the state of each call must be included. Such call state data may include the progress of the call and the health of the transport network carrying the call. Several existing standards describe this signalling data. This Recommendation describes the format for signalling data in the IP packets that carry the T1 information over DOCSIS channels.
- Operations – The synchronous transport systems put together by telecommunications carriers for T1 data have been refined over the years to become very reliable. But the manner by which a DOCSIS transport system performs this service differs considerably from the manner in which they are performed in the PSTN by telecommunications carriers. Operators of DOCSIS systems can alter their standard operating procedures to accommodate these differences. This Recommendation makes many suggestions for operators who must configure DOCSIS systems to carry emulated T1/E1 data as specified herein.

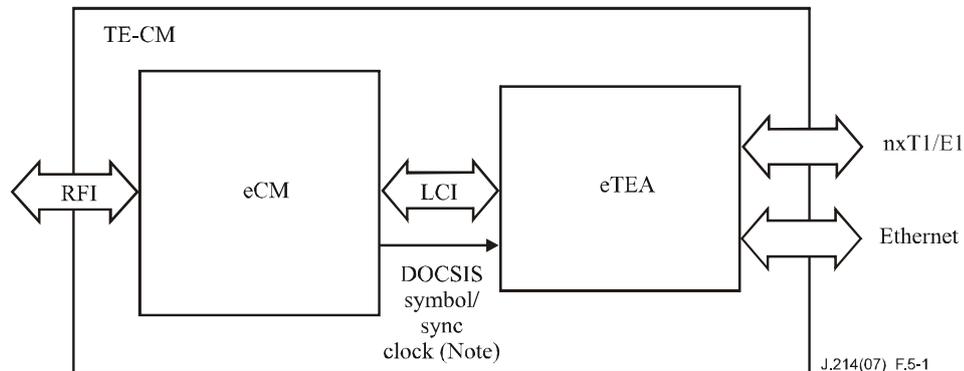
## 5.3 Devices/Interfaces

### 5.3.1 TE-CM

A TE-CM is a standard DOCSIS cable modem that contains an embedded TDM emulation adapter (eTEA) giving it the ability to offer T1/E1 services. A TE-CM may have architecture and interfaces similar to that shown in Figure 5-1. The power supply and other functionality that is out of scope are not shown. The elements/interfaces are described as follows:

- DOCSIS eCM – This is a standard DOCSIS CM conforming to DOCSIS 1.1, 2.0, 3.0 or greater. It must be able to carry a constant bit rate service flow carrying packets at bit rates sufficient to handle T1/E1 data, management and signalling as defined herein. It also contains [ITU-T J.126] functions sufficient to support eTEA and Ethernet functionality. Per DOCSIS, it has both RF and Ethernet interfaces. The embedded DOCSIS CM chipset provides clocking to the eTEA recovered from the DOCSIS symbol clock or DOCSIS SYNC message as defined in clause 6.4.
- eTEA – The TE-CM contains an eTEA that operates as an eDOCSIS eSAFE.
- T1/E1 Interface – This interface is a standard T1/E1 connector.

- Ethernet interface – This interface is a standard Ethernet connector (RJ45). It may, in future, provide the capability to exchange IP packets with eTEA or other TDM-E entities that may exist outside the TE-CM.
- Logical CPE interface – This is an unspecified interface between the eTEA and eCM.



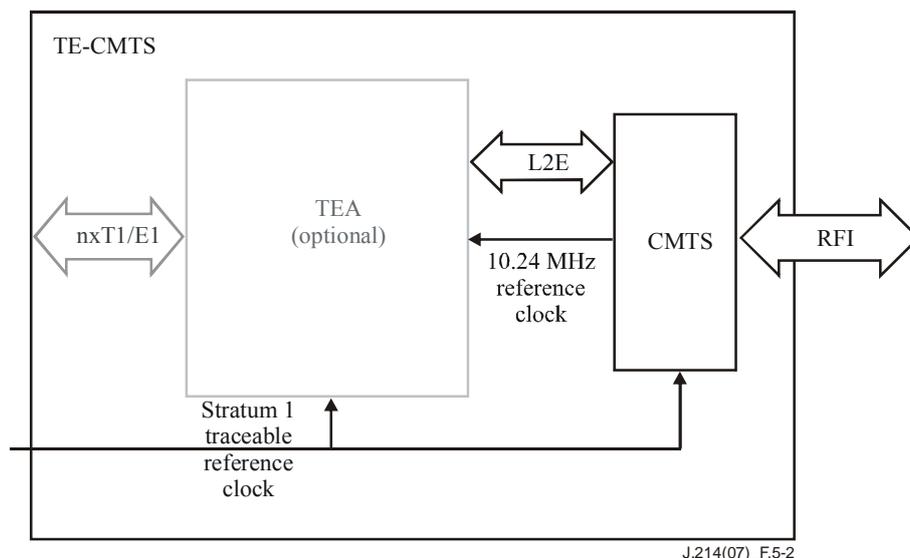
NOTE – [ITU-T J.126] applies to the eCM. The DOCSIS Symbol/Sync clock is Stratum 1 Traceable.

**Figure 5-1 – TE-CM**

### 5.3.2 TE-CMSTS

A TE-CMSTS is a standard DOCSIS CMTS that contains the necessary clocking functionality necessary to support a TE-CM and may also include a TDM emulation adapter (TEA) giving it the ability to terminate T1/E1 services. A TE-CMSTS may have architecture and interfaces similar to that shown in Figure 5-2. The power supply and other functionality that is out of scope are not shown. The elements/interfaces are described as follows:

- RF – This is the standard DOCSIS CMTS signal as transceived on the HFC network.
- DOCSIS CMTS – This is a standard DOCSIS CMTS conforming to DOCSIS 1.1, 2.0, 3.0 or greater. It must be able to carry a constant bit rate service flow carrying packets at bit rates sufficient to handle the IP pseudo wire traffic and management as defined herein. Per DOCSIS, it has both RF and IP interfaces. The CMTS may be an M-CMTS in which case the M-CMTS Recommendation defines compliant IP, RF and synchronization interfaces.
- Clocking unit – When the TEA is not integrated within the TE-CMSTS, at a minimum the clocking unit functionality must be implemented as part of the TE-CMSTS. A TE-CMSTS has either a [ITU-T J.211] or a BITS (building integrated timing supply) external synchronization interface as defined in clause 6.4. This is used to time and/or clock the DOCSIS symbol clock or DOCSIS SYNC message for a TE-CM to recover.
- TEA – A TE-CMSTS may contain a TEA and TDM interface(s).
- Ethernet interface – The Ethernet interface will carry IP traffic to and from the managed IP network.



**Figure 5-2 – TE-CMTS**

L2E is an unspecified Layer 2 Ethernet interface.

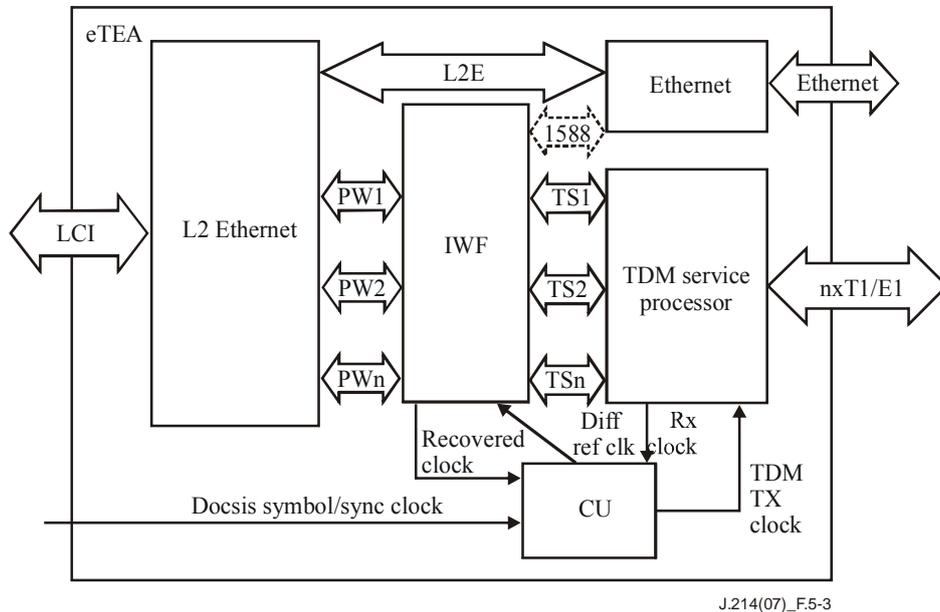
### 5.3.3 Embedded TDM emulation adapter (eTEA)

An Embedded TDM emulation adapter (eTEA) is an eSAFE device as specified by [ITU-T J.126]. An eTEA is embedded in a TE-CM.

An eTEA has a T1 or E1 interface connector and contains the following elements/interfaces:

- L2 Ethernet – This is the Layer 2 Ethernet element where IP Pseudo wires are transceived.
- Pseudo Wires (PW1, PW2, ..., PWn) – Pseudo wires are logical IP packet flows that contain encapsulated TDM traffic. Pseudo wires are terminated and generated by IWF.
- Interworking function (IWF) – The interworking function consists of a TSP data interface on one side, an IP Packet interface on the other side, and the functionalities to encapsulate TDM data into a pseudo wire in one direction and extract TDM data from pseudo wire in the other direction. Encapsulation and transmission may involve clocking from the CU.
- TDM service flows (TS1, TS2, ..., TSn) – TDM data associated with a pseudo wire and TDM interface. The TDM service flows may be multiplexed onto one or more TDM interfaces through the TDM service processor.
- TDM service processor – The TDM service processor connects and may multiply TDM service flows onto one or more TDM interfaces.
- Ethernet interface. This Customer facing Ethernet interface has the following uses:
  - to launch the optional 1588 local master timing function on the customer Ethernet network per clause 6.4.5;
  - for DOCSIS testing of an eCM;
  - for testing of clock recovery per clause 6.4.7.1;
  - for (out of scope) use to carry TDM data per clause 5.3.1.
- TDM interface – The TDM interface is one or more T1, E1 or higher rate interface as further defined in clause 3.
- Clocking unit (CU) – This element provides high quality and reliable frequency to the IWF and TDM service processor. The CU receives clocking from the eDOCSIS module. The eDOCSIS module through DOCSIS requirements must lock to the DOCSIS symbol clock

and DOCSIS SYNC message. The eDOCSIS module must provide one of these recovered clocking signals to the CU. Since a TE-CMTS is synchronized to a Stratum 1 traceable reference, the recovered clock from DOCSIS not only has the stability needed for DOCSIS operation, it also has stability needed for T1/E1 synchronization. Clause 6.4 describes several clocking modes that may include the CU interfacing with the TDM service processor and IWF.



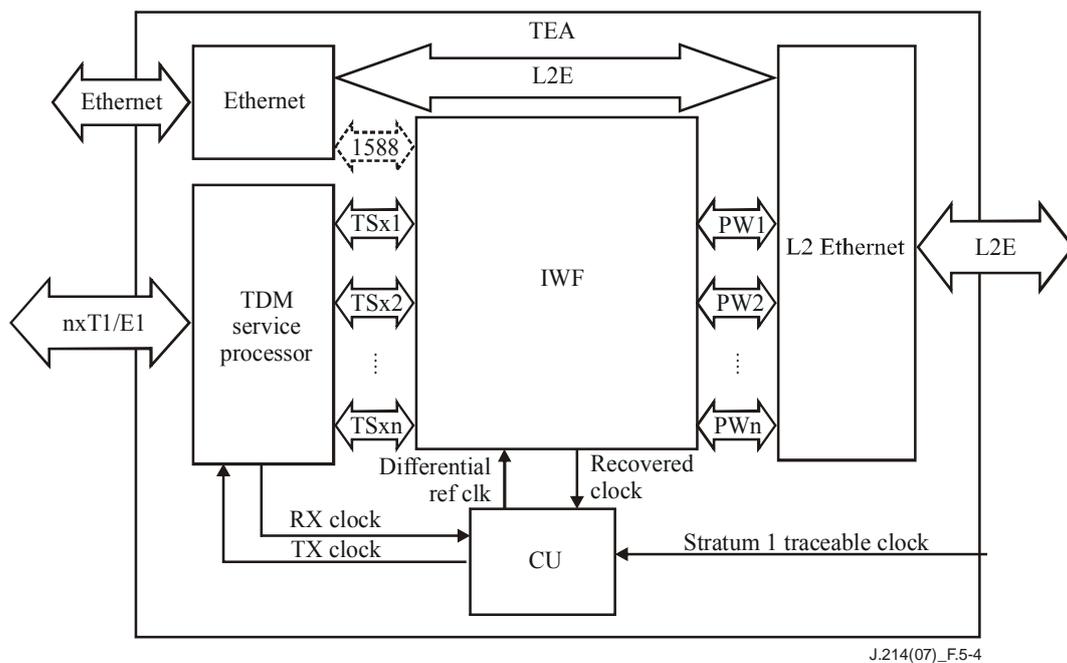
**Figure 5-3 – eTEA**

### 5.3.4 TDM emulation adapter (TEA)

A TDM emulation adapter (TEA) may be implemented in a TE-CMTS or other device unspecified in the network.

A TEA has a T1 or E1 TDM interface connector and contains the following elements/interfaces:

- L2 Ethernet – This is the Layer 2 Ethernet element where IP pseudo wires are transceived.
- Pseudo wires (PW1, PW2, ..., PWn) – Pseudo wires are logical IP packet flows that contain encapsulated TDM traffic. Pseudo wires are terminated and generated by IWF.
- Interworking function (IWF) – The Interworking function consists of a TSP data interface on one side, an IP packet interface on the other side, and the functionalities to encapsulate TDM data into a pseudo wire in one direction and extract TDM data from pseudo wire in the other direction. Encapsulation and transmission may involve clocking from the CU.
- TDM streams (TS1, TS2, ..., TSn) – TDM data associated with a pseudo wire and TDM interface. The TDM streams may be multiplexed onto one or more TDM interfaces through the TDM service processor.
- TDM service processor – The TDM service processor connects and may multiplex TDM streams onto one or more TDM interfaces.
- TDM interface – The TDM interface is one or more T1, E1 (or higher rate) TDM interface.
- Clocking unit (CU) – This element provides high quality and reliable frequency to the IWF and TDM service processor. The CU receives clocking from a Stratum 1 traceable reference through either a DTI or BITS interface, and a recovered clock from the clock recovery unit within the IWF. Clause 6.4 describes several clocking modes that may include the CU interfacing with the TDM service processor and IWF.



**Figure 5-4 – TEA**

## 5.4 Architecture

A TDM-E service can be deployed in several architectures. In any architecture, the TDM-E service has two end points that each contain a TEA or eTEA (TEA peers). Each device presents a TDM interface to the customer or network and a pseudo wire connecting the TEA peers logically across the IP network.

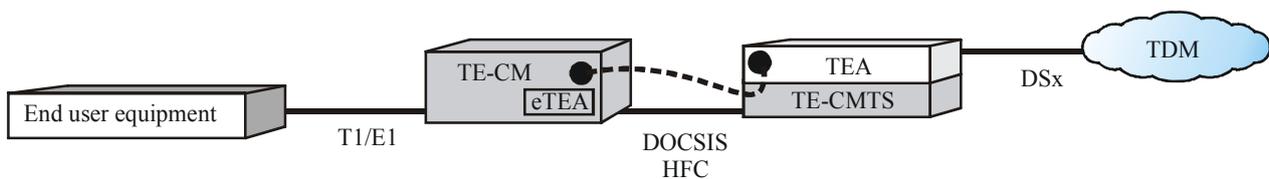
Three general architectures exist based on the existing market for T1 and E1 services. In all three architectures, one end of the TDM-E service originates in a TE-CM containing an eTEA. The three architectures differ based on where the other end of the service is terminated and thus where the other TEA/eTEA exists.

There are a variety of existing T1/E1 services including wireless backhaul, private line, PBX extension, PBX PSTN access, etc. Comprehensive T1/E1 service may include a combination of these three architectures and may include partnering with other operators for part of the implementation.

### 5.4.1 TE-CM to TE-CMTS w/TEA

In this architecture, the TDM-E service originates in a TE-CM (containing an eTEA) and terminates at the TE-CMTS which contains a TEA. Figure 5-5 shows this architecture with the pseudo wire flow shown as the dotted line. This is not the termination of the T1 or E1 circuit, just the two ends of the TDM-E circuit emulation. The T1 or E1 service will continue from the CMTS into the traditional TDM network.

This architecture places delay and QoS requirements on the DOCSIS network, but minimizes the impairments that the IP network may place on the TDM-E service. This architecture, however, requires TDM transport facilities at the hub/headend locations where the CMTS is located which may not be feasible or cost effective for some operators that prefer or require the TDM-E service to have a more centralized termination.



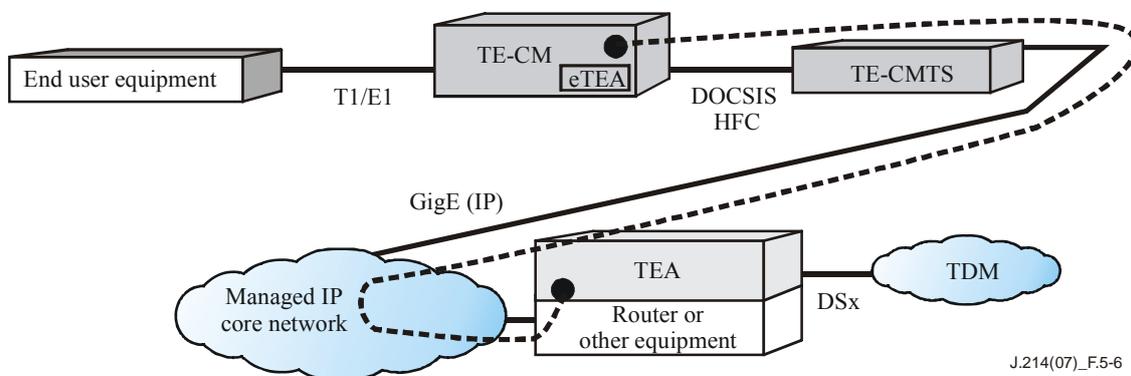
J.214(07)\_F.5-5

**Figure 5-5 – T1 to hub architecture**

#### 5.4.2 TE-CM to centralized TEA

In this architecture, the TDM-E service originates in a TE-CM and terminates in a centralized router or other unspecified device that contains a TEA. The TEA may exist in a router, gateway, switch or other device which is out-of-scope for this Recommendation with the exception of the TEA functionality. Figure 5-6 shows this architecture with the pseudo wire flow shown as the dotted line. This is not the termination of the T1 or E1 circuit, just the two ends of the TDM-E circuit emulation. The T1 or E1 service will continue from the CMTS to its termination within traditional TDM network.

This architecture places delay, synchronization and QoS requirements on the TE-CMTS, which in this case does not contain a TEA. This architecture places delay and QoS requirements on the DOCSIS and core managed IP network. While this architecture requires a slightly more complex provisioning of the network, it centralizes the termination of the TDM-E circuit emulation in the core. This means there are fewer TDM-E devices to manage and that the TE-CMTS does not have an integrated TEA. This simplifies the TE-CMTS and reduces or eliminates modifications to existing installed CMTSs. This architecture eliminates the need for TDM transport facilities at each hub/headend since the pseudo wires will pass through the TE-CMTS (as IP packets) and be routed over the existing IP transport to the core.



J.214(07)\_F.5-6

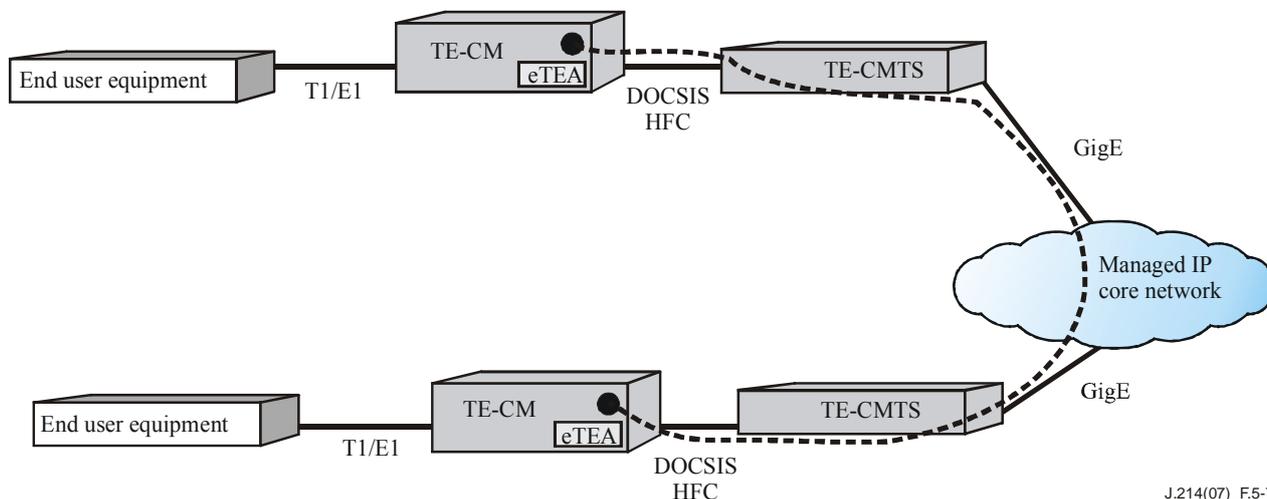
**Figure 5-6 – T1 to TDM core, IXC or PSTN**

#### 5.4.3 TE-CM to TE-CM through TE-CMTSs and core

In this architecture, the TDM-E service originates in a TE-CM (containing an eTEA) and in a second TE-CM (containing an eTEA). Figure 5-7 shows this architecture with the pseudo wire flow shown as the dotted line. In this architecture, the T1 service is end-to-end with transport of the circuit through the managed IP core network. If the T1 circuit requested by the customer is accessible by a single hub/headend, a single TE-CMTS may be utilized with the pseudo wire being routed locally.

This architecture places delay, synchronization and QoS requirements on the TE-CMTS, which in this case does not have a TEA. This architecture places delay and QoS requirements on the DOCSIS and core managed IP network. While this architecture requires a more complex

provisioning of the network, it can be deployed with minimal upgrades to the existing installed CMTSs and does not require any modifications to the core equipment. This architecture eliminates the need for TDM transport facilities at each hub/headend since the pseudo wires will pass through the TE-CMTSs and existing IP transport in the core between the two TE-CMs. There are, however, service limitations imposed by this architecture. This architecture only enables a provider to offer private line T1/E1 services between two customer sites. This architecture does not enable T1/E1 services to the PSTN or any other TDM network.



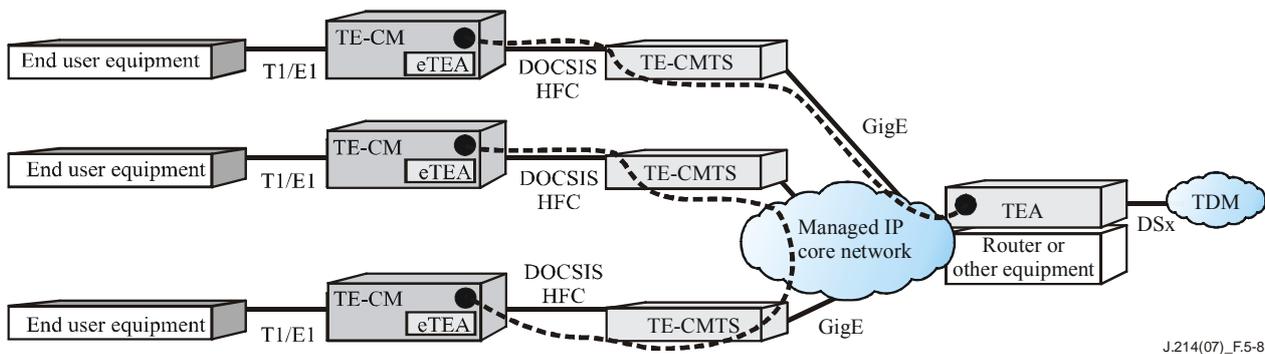
J.214(07)\_F5-7

**Figure 5-7 – T1 private line between two customer sites**

#### 5.4.4 Other architecture considerations

For many operators, a complete T1/E1 service offering will require a mixture of these three architectures. In most cases, this will be a mixture from clauses 5.4.3 and either clause 5.4.1 or 5.4.2. Figure 5-8 depicts a potential hybrid architecture.

An operator may also choose to partner with other operators creating exchange agreements where a pseudo wire may flow over both of the managed IP networks and terminate in a TE-CM in each of their networks. Moreover, an operator may choose to partner with another operator to offer T1/E1 services outside their network nationally or internationally. In most cases, this will require the operator to terminate and aggregate the TDM-E circuit emulation for handoff as a high rate TDM interface, like a DS3. This would also be the case if an operator chooses to offer wireless backhaul services. The operator would need to aggregate multiple Pseudo Wire IP flows from several TE-CMs, terminate the TDM-E circuit emulation and handoff to the wireless operator in a high speed aggregate TDM DS3 type interface. Lastly, an operator could choose the TEA defined in this Recommendation to transport TDM traffic across the managed IP core network between two routers or other devices. The details of this are out-of-scope for this Recommendation; however, it is possible with this technology.



**Figure 5-8 – Comprehensive private and public T1/E1 services**

### 5.4.5 M-CMSTS & CMSTS architecture considerations

The Modular-CMSTS (M-CMSTS) architecture offers many advantages when considering a TDM-E service. Since a T1 or E1 service has constant symmetric bandwidth of 1.5 Mbit/s to 2 Mbit/s, a traditional CMSTS architecture may not scale as well as an M-CMSTS. The M-CMSTS architecture decouples the upstream and downstream into separate devices that can be scaled independently. Since TDM-E, residential broadband data service, VoIP and other services all utilize the DOCSIS network this independent scalability offers greater economics and flexibility as various services are deployed.

In addition to the scalability, Modular-CMSTS requires the DOCSIS timing interface ([ITU-T J.211]). This is used for modular-CMSTS operation, but can also be used to support TDM-E synchronization requirements as described in clause 6.4.

Existing CMSTS architectures may be utilized for TDM-E as well as the M-CMSTS devices that are inherently compliant. Existing CMSTS may or may not have been designed with TDM-E in mind and may need some modifications. The potential for a CMSTS to be made compliant to TDM-E as a TE-CMSTS is vendor and platform dependent. The CMSTS will need to have a synchronization interface and support QoS requirements in order to support advanced clocking modes beyond adaptive clock recovery. This may not require any modifications for some vendors, for others rather minor changes and for some it may not be possible on existing platforms. Moreover, some vendors may be able to offer upgrades to existing CMSTS platforms to not only include TE-CMSTS functionality, but also an integrated TEA.

## 5.5 Network environment

The architecture shown in clause 5.4 serves as the basis for explaining how DOCSIS can be used to carry T1/E1 using TDM-E. The architectures do not, however, show details of the environment; details that are just as important as the devices. To adequately explain these extended architectures, it is important to outline the environmental details related to clocking and QoS.

### 5.5.1 Clocking

#### 5.5.1.1 Background

T1/E1 signals are designed to simultaneously connect multiple calls over a shared wire using time domain multiplexing (TDM). This imposes certain restrictions on the T1/E1 clocks used at both ends of the connection.

- Frequency – To make sure that each TDM circuit stays properly connected, the clocks at both ends of the circuit must be at the same frequency. This ensures that both ends of the TDM connection continue to use the same shared data time-slot, without either end slowly slipping into the next time-slot. In other words, the data rate in and the data rate out must be precisely aligned and not deviate. The responsibility for maintaining the same frequency is

shared by the two TEAs and the clock distribution system described below in clause 5.5.1.2.

- Synchronization – To make sure that each DS0 frame is correctly multiplexed into the TDM circuit, the peer TEAs must both be synchronized to the frame of the incoming T1/E1 signal. The responsibility for maintaining frame synchronization is shared by the peer TEAs; they should be configured in a similar manner.

### **5.5.1.2 Clock distribution environment**

There are several options available to ensure that the peer TEAs operate at the same clock frequency. From wherever the source clock exists, the frequency information about that clock must be distributed to each TEA. A few different distribution paths and methodologies exist over which a TEA or eTEA can obtain this clock information.

#### **5.5.1.2.1 Direct clocking**

Direct clocking methodology means the TEA is receiving its clocking directly from a Stratum 1 traceable reference source. This may be one of two technologies: [ITU-T J.211] or BITS.

- DTI – The DOCSIS timing interface Recommendation is specified as part of the DOCSIS M-CMTS family. [ITU-T J.211] describes a Client/Server architecture for time and frequency distribution. An M-CMTS device, CMTS or any device containing a TEA may implement a DTI client interface for connection to a DTI server. This will enable it to not only receive timing for TDM-E service, but also DOCSIS. The DTI server being used to time a device containing a TEA must implement the External Reference Mode of [ITU-T J.211] in order to comply with TDM-E. There are no additional requirements for the DTI Client.
- BITS – BITS is a generic term used to describe frequency distribution of Stratum 1 clocks in a TDM, SDH or PDH network. BITS is described in detail in ANSI T1.101-1999. In a typical deployment, a primary reference source (PRS) is containing a Stratum 1 clock or GPS (Stratum 1 traceable) connection is distributed in a Central Office, Headend or hub site directly from the PRS or through a building integrated timing supply shelf (BITS shelf). In some cases, the PRS and BITS shelf may be integrated into a single shelf called a synchronization supply unit (SSU). A network element that needs frequency synchronization will typically have a primary and secondary frequency input, typically labelled BITS input. The BITS input may have a variety of physical interfaces including T1, E1, or RS422. There are a few quality levels of BITS connections depending on the PRS and BITS topology and architecture. The same Recommendations [ITU-T G.823] and [ITU-T G.824] that govern customer T1/E1 synchronization quality also describe the PRS BITS quality used for direct clocking.

#### **5.5.1.2.2 Network clock recovery (NCR)**

Network clock recovery methodology means the TEA is receiving its clocking over the transport network (in our case DOCSIS) from a device (in our case a TE-CMTS) that has direct clocking. A device using NCR is thus traceable to a Stratum 1 clock, but has slightly lower quality clocking than a device that is directly clocked using DTI or BITS. Since DTI and BITS are both inter-office technologies, NCR is the best methodology to provide clocking to a modem. There are two methods for a TE-CM to implement NCR: DOCSIS Symbol Clock recovery and DOCSIS SYNC Message Recovery.

- DOCSIS symbol clock (SC) – The TE-CM can receive clock information encoded into the DOCSIS symbol clock. The CMTS or M-CMTS connected to the TE-CM has a clocking interface to which the DOCSIS master clock is locked. This ensures that the DOCSIS symbol clock not only meets DOCSIS requirements, but also the T1/E1 clocking requirements. The TE-CM can then provide the recovered DOCSIS symbol clock to the

eTEA. This is the most reliable and accurate method of clock distribution. One example of how this might be done is described in clause 6.4.2.2.

- DOCSIS SYNC Message Recovery – The TE-CM can receive clock information encoded in a DOCSIS SYNC Message. The CMTS or M-CMTS connected to the TE-CM has a clocking interface to which the DOCSIS master clock is locked. This ensures that the DOCSIS SYNC message not only meets DOCSIS requirements, but also T1 clocking requirements. The TE-CM can then provide a frequency reference to the eTEA disciplined by the arrival of the DOCSIS SYNC Message. One example of how this might be done is described in clause 6.4.2.3.

Direct clocking and NCR are the two most widely deployed and accurate timing methodologies deployed in communication networks today. They both are well documented and specified by the ITU and ATIS to ensure reliable and interoperable equipment and services globally. Historically, these two methodologies have been bedrock for TDM, SDH, PDH and ATM networks. However, IP/Ethernet networks are just now considering how to provide direct clocking and network clocking. DOCSIS is the first widely deployed access technology that can transport IP traffic and Stratum 1 traceable synchronization. TDM-E leverages this capability to provide a more robust IP circuit emulation service than specified by other existing Circuit Emulation standards that do not have direct clocking or NCR capabilities at this time.

#### **5.5.1.2.3 Prime differential clock recovery (PDCR)**

Prime differential clock recovery methodology leverages NCR enabling peer TEAs to precisely measure the customer's T1/E1 clock and transport these measurements over the network. PDCR may be very useful for some data or cellular applications where Stratum 1 quality is needed; however, the customer has a different reference clock in their equipment.

#### **5.5.1.2.4 Adaptive clock recovery (ACR)**

Adaptive clock recovery methodology uses the customer clock to discipline a TEA (or eTEA). At the second TEA, the packet arrival rate is used to derive the original clock. In this methodology, complex algorithms and hardware are needed to filter out the jitter and wander to meet T1/E1 clocking standards. Various tradeoffs regarding cost, reliability and interoperability should be considered when using ACR. Startup time, instantaneous network delay, delay variation, packet loss, loading, and QoS all affect the clocking. The margin of safety is largely dependant on the QoS agreements that the service provider can offer.

When using ACR in the upstream direction, the quality of the recovered clock is not only affected by the IP network, it is affected by the upstream DOCSIS flow. Unsolicited grant service (UGS) is the best method to use in DOCSIS to ensure that the TDM-E service has the highest possible quality. The contention in the upstream A-TDMA or S-CDMA DOCSIS transmission will cause packet delay variation even with UGS. The effect of this will typically not have an effect on the data content or delivery; however, it may have effects on the quality of timing. To ensure high quality clocking is delivered to the customer direct clocking, NCR or PDCR should be used if available.

When using ACR across an IP network, the QoS must be provided end to end. This may impose unspecified restrictions on the combined IP network. Differential clocking (PDCR) does have some dependencies on the IP network, but is less restrictive than ACR because the actual clock is not being transmitted across the network. Rather each TEA is being timed from a Stratum 1 traceable source using NCR and the difference between the network clock and customer clock is being transmitted.

## 5.5.2 Quality of service (QoS)

### 5.5.2.1 Background

T1 services provide transport for constant bit rate (CBR) signals. Because the transport of a pseudo wire involves the sharing of resources, with the attendant requirements for reservations, IP data flows are inherently not CBR. There are two classical ways to make up for this deficiency:

- Buffering – Delay all of the IP data; buffer it into queues, and clock it out in a CBR manner. This can be done in the middle of the network IP path, but it is best done at the edge of the network, where the PW terminates in the IWF. The buffering is provided by the IWF and is configured therein. Clause 5.5.2.2.2 further describes buffering. While not strictly a 'QoS' consideration, the buffer size of the IWF must be well matched to the level of QoS so it can smooth out the IP data flow.
- QoS – Give the PW data priority over other types of data and tag it as CBR data. The IP network elements may provide QoS options allowing the operator to specify the proper priority.

### 5.5.2.2 QoS environment

#### 5.5.2.2.1 QoS

The principal requirement (for a DOCSIS transport system implementing TDM-E) is to guarantee CBR transport of pseudo wire packets. In the theoretical best case, every bit presented to the network is immediately sent through the network with low and consistent delay. In the practical case, the network cannot be quite that accommodating. It may be forced by contending users to buffer (delay) traffic now and then catch up later by transporting data at a higher rate. The QoS mechanisms provided by DOCSIS can limit the amount of buffering that occurs in the network. The buffer in the TEA must be more than a match for the network buffering, or the system will break down. The better the QoS afforded to pseudo wire packets, the better the TDM-E service will function. The operators should provision the downstream and upstream of a TE-CMTS and TE-CM for the delivery of constant bit rate PWs. In the upstream, this will likely mean UGS as further defined in clause 6.7. If the pseudo wire packets leave the DOCSIS network, they must be transported with equivalent QoS. Such methods are out of scope. The weakest link in the transport network will set the expectations for remedial buffering in the TEA.

#### 5.5.2.2.2 IWF buffering

The IWF has a configurable number of bytes to buffer each incoming pseudo wire packet stream. The IWF then performs interworking on the pseudo wire packets generating a TDM stream to the TSP.

#### 5.5.2.2.3 Latency vs buffering errors

There is a tradeoff between the size of the buffer and the statistical chance of errors. At any level of QoS, there will be a statistical spread of packet delay. Each packet may get delayed a different amount. For the most part, a large delay would be more rare than a smaller delay. The IWF buffer must be able to accommodate almost every possible packet delay and packet delay variation. It must do so without overflow or underflow, both of which will create data errors in the T1/E1 service noticeable to the customer as BER, slips or LoS. To play it safe, the PW could always be configured with a very large buffer. This would minimize data errors. Unfortunately, many of the services carried over T1/E1 connections can only tolerate certain maximum delay times (out of scope). This max delay time puts a cap on the allowable size of the IWF buffer in that application. The cap on buffer size, in turn, places requirements for QoS on the DOCSIS network. Everything must be well matched. The goal is to simultaneously minimize the size of the buffer on each PW and stay within the statistical bounds allowable for the probability of buffer errors. Operational guidelines are out of scope for this Recommendation.

## **6 Technical specification**

### **6.1 General requirements**

TDM-E circuit emulation service provides emulation of T1 or E1 or nxDS0 service over a DOCSIS network. A system providing this service may consist of a TDM emulation capable CMTS (TE-CMTS), a TDM emulation cable modem (TE-CM), and possibly another router on the network side that contains a TDM emulation adapter (TEA).

#### **6.1.1 clock unit (CU)**

A CU gathers, transforms, and redistributes clocking information within a given device. There are many options as to how this may be done. The requirements for the sources of the clocking information, the transformation methods, and the redistribution methods are documented in clause 6.4.

#### **6.1.2 pseudo wire (PW)**

A pseudo wire (PW) is a flow of encapsulated TDM data within IP packets. A PW always flows between TEA peers across DOCSIS and a packet switched network.

#### **6.1.3 Interworking function (IWF)**

An IWF encodes TDM flows into IP pseudo wire flows and decodes IP pseudo wire flows into TDM flows.

An IWF **MUST** support at least one PW. An IWF **MAY** support multiple PWs. If the IWF supports multiple PWs, the configuration of each PW may be independent.

An IWF **MUST** have a configurable-sized input buffer to receive each incoming pseudo wire flow large enough to hold at least 10 ms of data at the PW line rate. For certain network conditions, larger buffers may be needed. An IWF **SHOULD** have a configurable-sized input buffer to receive incoming pseudo wire flows large enough to hold 50 ms of data at the PW line rate. The IWF should have internal control algorithms that will make sure the buffer operates near the centre of its range during normal transport (after the service is provisioned and initialized) for each PW. This will minimize buffer overflow and underflow and will help minimize the size of buffer that is needed. The buffer size should be configured to the minimum effective size to minimize latency. The algorithm necessary to control the buffer is out of scope.

An IWF **MUST** support TDM-IP interworking as defined in clause 6.2.

An IWF **MUST** support structure agnostic encapsulation as defined in clause 6.3.1.

An IWF **MUST** support structure-aware encapsulation as defined in clause 6.3.2.

An IWF **MUST** support TDM Application signalling defined in clause 6.3.2.1.

An IWF **SHOULD** support PDCR as defined in clause 6.4.3.

An IWF **MAY** support ACR as defined in clause 6.4.4.

If ACR or PDCR are supported for encapsulated PWs, the IWF **MUST** be capable of gathering clocking information from those PWs and supplying it to/from the CU.

#### **6.1.4 TDM service processor (TSP)**

A TSP **MUST** have a TDM interface for each TDM flow it can simultaneously support; the interfaces may be removable if they are only used during testing.

A TSP **MUST** accept clocking from the CU.

The TSP **MUST** provide the TDM Receive clock to the CU.

If the TSP TDM interface is a T1, it MUST comply with:

- [ITU-T G.703];
- [ITU-T G.704];
- [ITU-T G.824] as further defined in Table 6-1;
- Other National standards that are appropriate, such as [T1.403] or [TR-62411].

If the TSP TDM interface is an E1, it MUST comply with:

- [ITU-T G.703];
- [ITU-T G.704];
- [ITU-T G.706];
- [ITU-T G.732];
- [ITU-T G.823] as further defined in Table 6-1.

[ITU-T G.823] and [ITU-T G.824] differ in the requirements placed on a 'Traffic Interface' and 'Synchronization Interface' as defined therein. This Recommendation places normative requirements on TDM interfaces to conform to these ITU-T Recommendations, and Table 6-1 below is required to fully specify which clauses of the ITU-T Recommendations apply. Table 6-1 is used to specify different requirements for different TDM clocking methods which are defined in the clauses cited in the table.

A TSP TDM T1 'Traffic' interface MUST comply with:

- [ITU-T G.824] Table 1 for 1544 kbit/s digital rate interfaces;
- [ITU-T G.824] Table 2.

A TSP TDM T1 'Synchronization' interface MUST comply with:

- [ITU-T G.824] Table 1 for 1544 kbit/s digital rate interfaces;
- [ITU-T G.824] Table 5, Figure 3;
- [ITU-T G.824] Table 6, Figure 4.

A TSP TDM E1 'Traffic' interface MUST comply with:

- [ITU-T G.823] Table 1 for 2048 kbit/s digital rate interfaces;
- [ITU-T G.823] Table 2, Figure 1.

A TSP TDM E1 'Synchronization' interface MUST comply with:

- [ITU-T G.823] Table 5 for PDH synchronization;
- [ITU-T G.823] Table 12, Figure 10;
- [ITU-T G.823] Table 13, Figure 11.

**Table 6-1 – G.823 and G.824 clock requirements**

<b>Clocking method</b>	<b>Clause</b>	<b>Traffic/Synchronization</b>
Direct	6.4.1	Synchronization
NCR	6.4.2	Synchronization
PDCR	6.4.3	Synchronization
ACR	6.4.4	Traffic

A TSP interface that complies with [ITU-T G.823] or [ITU-T G.824] SHOULD comply as a 'Synchronization Interface'.

### **6.1.5 TDM emulation adapter (TEA)**

A TEA MUST contain an IWF.

A TEA MUST contain a TSP.

A TEA MUST contain a CU supporting a synchronized clock input to support NCR or direct clocking modes.

A TEA SHOULD contain an IWF supporting PDCR mode.

A TEA MAY contain an IWF supporting ACR.

A TEA SHOULD indicate when the CU, IWF or TSP are not locked to a valid clock, including periods when it is acquiring a clock signal using ACR (clause 6.4.4), PDCR (clause 6.4.3), or SC (clause 6.4.2.2).

A TEA must be associated with a static IP address.

A TEA MUST have an LCI interface capable of accepting IP packets.

If a TEA contains an IWF that contains multiple PWs:

- the TEA MUST be capable of supporting UDP addressing per clause 6.2.2.
- the TEA MUST meet this Recommendation on all TDM interfaces and PWs simultaneously.
- the TEA MUST be configurable for each PW without disturbing configuration on any other PW (exclusive of intentionally shared parameters).

A TEA MUST make available every active TDM interface of the embedded TSP.

A TEA MUST support the provisioning of the CU, the TSP, the IWF, and the PWs it contains.

### **6.1.6 embedded TDM emulation adapter (eTEA)**

An eTEA MUST meet the normative requirements of a TEA.

### **6.1.7 TDM emulation cable modem (TE-CM)**

A TDM emulation cable modem (TE-CM) is a cable modem with additional functionalities to provide TDM emulation services.

A TE-CM MUST support either DOCSIS 1.1 [ITU-T J.112], DOCSIS 2.0 [ITU-T J.122] or greater.

A TE-CM MUST contain an eCM as defined in [ITU-T J.126].

A TE-CM MUST contain an eTEA as defined in clause 6.1.5.

A TE-CM MUST make available every TDM interface made available by its embedded eTEA.

A TE-CM MUST support a CPE interface that includes 10BASE-T capability. A TE-CM SHOULD have an interface with 100BASE-T capability. The 10BASE-T or 100BASE-T interface MAY be compliant to clause 6.4. One example of an acceptable industry interface is CMCI.

A TE-CM MUST contain a CU that accepts a synchronized clock input in order to support NCR mode. This allows the TE-CM to lock to the DOCSIS ([ITU-T J.122]) symbol clock or DOCSIS SYNC message and provide it to the TSP and IWF as described in clause 6.4.

A TE-CM MAY contain an Ethernet timing service based on [IEEE 1588] as outlined in clause 6.4.5.

### **6.1.8 TDM emulation CMTS (TE-CMTS)**

A TDM emulation cable modem termination system (TE-CMTS) is a CMTS (or M-CMTS) with additional functionalities to enable TDM emulation services.

A TE-CMTS MUST support either DOCSIS 1.1, DOCSIS 2.0 [ITU-T J.122], or greater.

A TE-CMTS MUST contain a CU supporting direct clocking mode.

A TE-CMTS must be able to distribute clocking information downstream to the TE-CMs.

- A TE-CMTS SHOULD be able to lock the DOCSIS ([ITU-T J.122]) SYNC message to the CU as described in clause 6.4 and keep the DOCSIS SYNC Message compliant with clause 6.4.6.
- A TE-CMTS SHOULD be able to lock the DOCSIS ([ITU-T J.122]) symbol clock to the CU as described in clause 6.4 and keep the DOCSIS Symbol Clock compliant with clause 6.4.6.
- A TE-CMTS MUST be able to lock the symbol clock or the SYNC message to the CU as described in this clause. See clause 6.4.2.1 for further comments on this requirement.

A TE-CMTS MAY contain a TEA.

The TE-CMTS should provide low latency service to PW service flows in both the upstream and downstream directions. A TDM-E system should provide a round-trip latency of less than 14 ms, including the latency of the all equipment involved, including TE-CMTSs, TE-CMs and TEAs.

## 6.2 TDM-IP network interworking

This clause specifies the format of packet headers for PW packets. The format of the payload data for PW packets is specified in clause 6.3.

0				1				2				3										
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	Header
IPVER				IHL				IP TOS				Total Length				IP						
Identification										Flags		Fragment Offset				IP						
Time to Live						Protocol				IP Header Checksum				IP								
Source IP Address												IP										
Destination IP Address												IP										
Source Port Number						Destination Port Number						UDP										
UDP Length						UDP Checksum						UDP										
RES		L	R	M	RESERVED				Sequence Number				CW									
RTV	P	X	CC		M	PT				RTP Sequence Number				RTP								
Timestamp												RTP										
SSRC identifier												RTP										
Adapted Payload																						

**Figure 6-1 – Header format (Only IPv4 shown)**

The first five rows are the IP header (in the case of IPv4), the sixth and seventh rows are the UDP header. Row 8 is the TDMoIP control word. Rows 9 through 11 are the optional RTP header.

An IWF MUST implement the header fields (and payload) shown in Figure 6-1 in the order shown, with the possible exception of the RTP header which is optional for each PW.

### 6.2.1 IP header fields

The IWF MUST support IPv4. The IWF MAY support IPv6.

The IWF MUST be configurable for the IPv type for each PW. The IWF MUST use the same IPv type for transmission and reception of PW flows. An IWF SHOULD use the same IPv type for both PW data flows and management.

The IWF MUST discard datagrams with zero Time to Live.

The IWF must not fragment transmitted data so it MUST set the IPv4 IP Header Flags (3 bits) to 010.

### 6.2.2 UDP header fields

Source and destination port numbers (16 bits each). Each TDM-bound PW of a given IWF must be given a unique UDP destination port number chosen from the range of dynamically allocated and private UDP ports numbers (49152 through 65535).

An IWF MUST use the destination UDP port field to differentiate between multiple PW streams. The source UDP port MAY be used to detect misconnections. An example can be found in section 6.6.1 of [b-MEF 8].

The IWF MUST NOT be configurable for 'well known' UDP ports (0 through 1023) per [IANA UDP PORTS]. UDP length (16 bits) is the length in bytes of the UDP header and data.

UDP checksum (16 bits) is the checksum of the UDP/IP header and data. If an IWF does not compute the UDP checksum, it MUST set the UDP checksum to 0.

### 6.2.3 Control word header fields

Emerging IETF and ITU-T Recommendations may differ from this Recommendation in the use of the control word header. In particular, the control word field and the optional RTP field may be in different order within the packet. This Recommendation was written to avoid a problem where optional [IETF RFC 3711] encryption would obliterate the control word if implemented. This Recommendation moves the control word outside the encryption range of [IETF RFC 3711].

The control word for MPLS PWE was moved ahead of the RTP header (in [IETF RFC 4553]) to avoid problems with [IETF RFC 4928] when MPLS is used. ECMP is a shortcut taken by MPLS routers that can cause continuous path shifting and a considerable increase in jitter. This is not a concern for this Recommendation because DOCSIS systems do not use MPLS internally. The reader is referred to section 4.3 of [IETF RFC 4553] and [IETF RFC 4928] for further reading. During transport of a PW over MPLS, users are advised to strip off the IP and UDP headers so the RESERVED field of the control word will prevent MPLS ECMP rerouting. This may create problems however for port labeled PWs. Alternatively, the headers can be left on if ECMP is turned off in the MPLS routing path(s). It is recommended that any transport of this PW data over MPLS be carefully done and follow the guidelines in [IETF RFC 4928] to avoid problems.

The control word header has the following fields:

RES (4 bits) is RESERVED. The IWF MUST set the four control word RESERVED bits to 0.

L (1 bit) is the Local TDM failure indicator bit.

- For structure-agnostic emulation:

The IWF MUST set the L bit when LoS is detected on the TDM interface associated with the PW.

The IWF MAY set the L bit when the AIS defect is detected on the TDM interface associated with the PW.

- For structure-aware emulation, the IWF MUST set the L bit when LoS, OOF or AIS defects are detected on a TDM interface associated with the PW.
- When the TDM fault is cleared, the IWF MUST clear the L bit.

R (1 bit) is the Remote Receive Failure bit:

- The IWF MUST set the R bit when it detects that the loss of packet state (LoPS) has been entered for the PW.

- The IWF MUST clear the R bit when it detects an exit from the loss of packet state for the PW.

### **M (2 bits)**

When the L bit is 0, the M bits (bit 6, bit 7) have the following meaning:

00 – Indicates regular TDM payload.

01 – Reserved.

10 – Reports the receipt of a defect such as RAI at the TDM input to the IWF. When this value is received by the peer IWF, RAI may be generated on the TDM interface according to the current configuration of RD action. This is applicable only to structure aware emulation.

11 – Indicates that the packet contains only the signalling sub-structure.

When L bit is 1, the M bits have the following meaning:

00 – Indicates there may not be valid TDM payload data in the packet.

01 – Reserved.

10 – Reserved.

11 – Reserved

RESERVED (8 bits). The IWF MUST set the 8 control word RESERVED bits to 0.

Sequence number (16 bits). The IWF SHOULD use this field to detect lost and reordered packets of each PW. The context of the sequence number is different for different values of the M bit. Signalling packets have their own sequence space. If RTP is not used, the IWF MUST process the sequence number (for each sequence space as selected by the M bits) as described in section 5.1 of [IETF RFC 3550]. See clause 6.2.4 below.

### **6.2.4 RTP header fields**

An IWF MAY support [IETF RFC 3550]. If the IWF supports PDCR, then it must support RTP per clause 6.4.3.

The RTP header field is optional and may be eliminated, but peer IWFs must be configured the same way. For reference, consult [IETF RFC 3550].

RTV (2 bits) is the RTP Version. The IWF MUST set the 2 RTV bits to 2.

P (1 bit) is the padding. The IWF MUST set the P bit to 0.

X (1 bit) is the header extension. The IWF MUST set the X bit to 0.

CC (4 bits) is the CSRC count. The IWF MUST set the 4 CC bits to 0.

M (1 bit) is the marker. The IWF MUST set the M bit to 0.

Accordingly, RTP header extensions, padding, and contributing synchronization sources are not used.

PT (7 bits) is the Payload Type. The IWF transmitting a PW MUST utilize a PT value provided from configuration (see clause 6.7.3 and "eTEA PW RTP Payload Type" in clause 6.7.1.10.1) to set the PT field for each packet transmitted for the PW. The IWF MAY use a configured PT value in incoming packets to detect malformed packets (see clause 6.7.3 and "eTEA PW RTP Peer Payload Type" in clause 6.7.1.10.1). See section 3 of [IETF RFC 3551] for more information on assignment of PT values in RTP. The following (decimal) PT value may be used in this Recommendation:

0 – For structure agnostic mode or if the PT is not used to detect malformed packets.

RTP sequence number (16 bits). The IWF MUST set and process the RTP sequence number as described in section 5.1 of [IETF RFC 3550]. The IWF MUST set the RTP sequence number identical to the sequence number in the control word.

RTP timestamps (32 bits). These are used for carrying optional timing information over the network. The IWF MUST generate timestamps for each PW in accordance with [IETF RFC 3550] if PDCR is supported.

SSRC (synchronization source) identifier (16 bits). The SSRC (synchronization source) identifier field in the RTP. This field should be used to indicate each unique clock domain and may be used for detection of misconnections.

### 6.2.5 RTCP extended report (XR) option

[IETF RFC 3611] specifies the method by which RTCP headers can be used as an in-band conduit for data that is pertinent to management, operation and performance. In this particular application, the principal benefit is that operators may be able to retrieve management information from remote TEAs by simply querying the local peer TEA. In addition, since the XR signalling data is in-band, it is possible to make one-way and round-trip latency and latency variation measurements.

- An IWF MAY support [IETF RFC 3611].
- An IWF should not support, or should not be configured to use SDP as it is applied in [IETF RFC 3611]. Peer PWs must be configured similarly for inter-operation using [IETF RFC 3611].
- An IWF that supports [IETF RFC 3611]:
  - MUST support XR Block Types 4 and 5 and be configurable to disable the sending of each Block Type. An IWF can use the Receiver Reference Time Report Block (Block Type 4) and the DLRR Report Block (Block Type 5) to make and report round-trip latency measurements.
  - MUST use the method outlined in [IETF RFC 3611] section 4.5, to report latency measurements. Further informative text may be found in [MLDA] and [IETF RFC 3550] Figure 2. The IWF transmitting Block Type 4 MAY substitute any timestamp in place of the NTP timestamp specified for Block Type 4 as long as it can adequately compute the round-trip delay from the echoed information received in the received Block Type 5 report.
  - MUST issue Block Type 4 and 5 reports at least every 10 minutes if they are enabled.
  - MUST NOT support any XR Block Type that it cannot disable or report as supported.
  - SHOULD be configurable to limit the total size of the RTCP header including all XR extension blocks. This may help prevent DoS attacks.
  - MUST be configurable as to the rate of transmission of each Block Type. This will allow operators to accurately predict data bandwidth.

It may be desirable to send XR messages on a regular basis. XR messages should be sent with UGS-AD scheduling and care should be taken to make sure the XR packets travel the same physical path as the payload packets.

### 6.3 Payload formats

This clause specifies the format of payload data for PW packets.

The IWF MUST preserve the order of payload bytes that correspond to their order in the TDM circuit.

The IWF MUST follow network bit order when creating packets from the TDM data and vice versa for each PW.

All packets belonging to a given PW in both directions must carry the same number of payload bytes. Accordingly, the IWF MUST process received PW packet payloads based on its configuration for the number of payload bytes used in transmitting PW packets.

The IWF MUST support a configurable packet payload size for each PW. The configuration of the packet size must remain unchanged for the lifetime of the PW. The total size of a PW packet for a specific PW flow must not exceed path MTU between the pair of IWFs.

An IWF MUST NOT support payloads smaller than 20 bytes.

Except for the possible presence of signalling packets (per clause 6.3.2), PW flows do not contain specific bits describing the data organization as being 'Structure Agnostic' or 'Structure Aware' as outlined in the following subclauses. This information is contained in the configuration of the pair of IWFs.

### **6.3.1 Structure-agnostic transport**

Structure-agnostic transport completely disregards any TDM structure and simply transports all data received from the TDM network in fixed length packets. No byte or frame alignment is implied. However, network bit order must be followed when creating packets from the TDM data and vice versa.

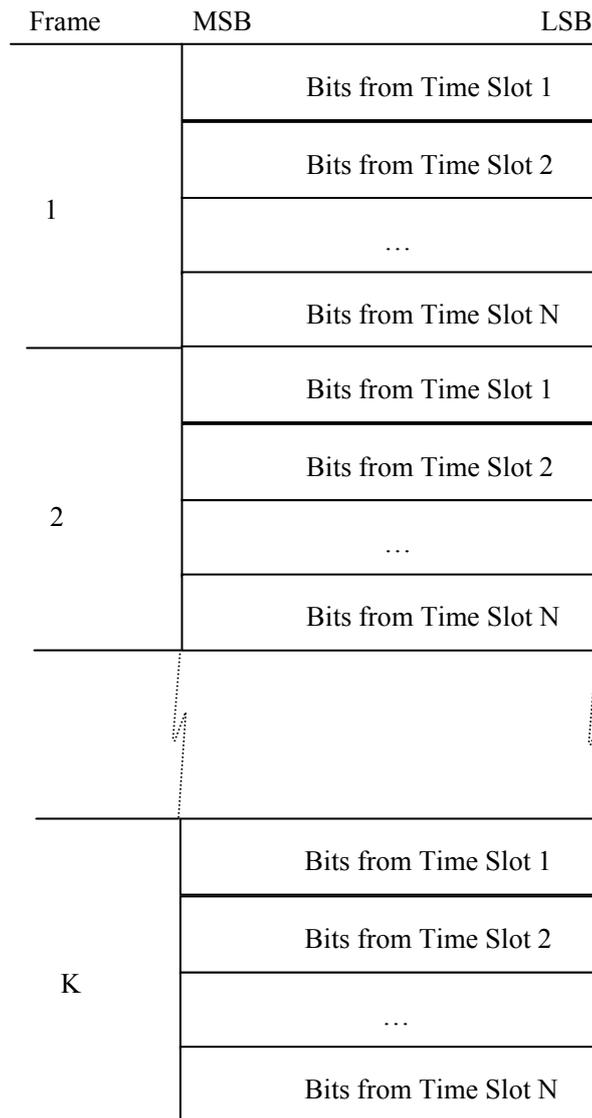
An IWF MUST be capable of supporting at least the following default payload sizes:

- E1 – 256 bytes.
- T1 – 192 bytes.

### **6.3.2 Structure-aware transport**

Structure-aware transport removes the structure overhead associated with the TDM data at the ingress IWF and regenerates it at the egress IWF. Only the payload extracted from the TDM frames is carried via the PW across the packet switched network. Structure-aware transport allows channelized/fractional T1/E1 operation.

In order to preserve the integrity of the TDM structure, for each PW the IWF MUST encode all non-signal packets to carry a fixed amount of data corresponding to the integral multiple (K) of TDM frames for which it is provisioned. Thus, the packetization delay for structure aware transport is  $K \times \text{frame interval}$  (i.e.,  $K \times 125 \mu\text{s}$ ). The resulting payload format is shown in Figure 6-2 below.



**Figure 6-2 – Payload format for structure-aware transport**

### 6.3.2.1 Signalling transport

TDM applications exchange signalling information in addition to TDM data. The typical example is telephony applications that exchange their state (i.e., off-hook/on-hook) in addition to TDM data carrying PCM-encoded voice.

When delivering the TDM services over a structure-agnostic pseudo wire, it is not necessary to intercept or process CE signalling. Signalling is embedded in the packetized TDM data stream, and hence it is carried end-to-end across the pseudo wire.

With structure-aware emulation, transport of common channel signalling (CCS) may be achieved by carrying the signalling channel together with the emulated service (i.e., channel 23 for DS1, or channel 16 for E1). However, channel associated signalling (CAS) (i.e., DS1 Robbed Bit Signalling or E1 CAS) requires knowledge of the relationship of the timeslot to the trunk multi-frame structure. This is indicated by the framing bits, which may not be preserved by  $N \times 64$  kbit/s basic service.

This clause describes a generic method for extending the  $N \times 64$  kbit/s basic service by carrying CAS signalling for each pseudo-wire in separate signalling packets that is independent of the TDM circuit type. It may be used in situations where the individual 64 kbit/s channels are selected from multiple TDM circuits, or picked off a TDM bus rather than from a specific TDM circuit. It also saves bandwidth, since only changes in the CE application state are carried.

### 6.3.2.2 PW signalling frames

The generic format of the PW signalling frames corresponds to the format shown in Figure 6-3. In DOCSIS systems, these frames are likely to be sent in a UGS-AD flow. The following additional requirements apply:

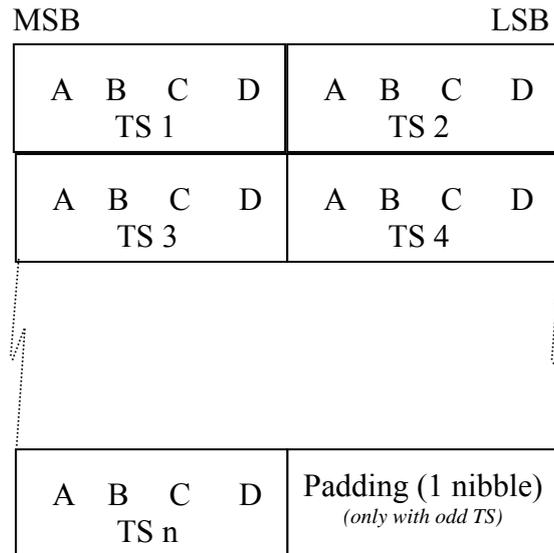
- An IWF MUST ensure that the following items are consistent between the PW data frames and their associated signalling frames:
  - Usage of the optional RTP header (that is, either both use it or both do not use it).
  - Destination IP Address and port.
  - Source IP Address.

The two types of frames differ only in the value of the M bits (signalling frame have a value of 'b11) and ToS bits (these are configurable separately for each frame type).

- An IWF MUST use a separate sequence number space for the PW data frames and their associated signalling frames.
- If an IWF uses the [IETF RFC 3550] header:
  - The IWF MUST use a payload type value provided from configuration (see clause 6.7.3 and "eTEA PW RTP CAS Signalling Payload Type" in clause 6.7.1.10.1) to set the PT field for associated signalling frames. The IWF MUST reject a configuration that assigns the same payload type value for data frames and signalling frames. The IWF MAY use a configured PT value in incoming packets to detect malformed packets (see clause 6.7.3 and "eTEA PW RTP CAS Signalling Peer Payload Type" in clause 6.7.1.10.1).
  - The IWF MUST assign identical timestamp values for the data and associated signalling frames at any given time.

NOTE – This enables synchronization of the signalling and data information using the standard RT-based mixing procedures described in [IETF RFC 3550].

- The IWF MUST assign the *i*th nibble of the payload the current "ABCD" value of the CAS signal corresponding to the *i*th timeslot of the PW as shown in Figure 6-3.
- The IWF MUST assign to the payload of each signalling frame an integral number of octets with at most padding of a single 4-bit nibble.



**Figure 6-3 – Signalling sub-structure**

The IWF MUST send PW packets with the signalling sub-structure as payload on a signalling change event including:

- Setup of the PW.
- A change in the signalling state of any of the 64 kbit/s channels of the PW.
- The loss of frames defect has been cleared.
- The remote loss of frames (R-bit) indication has been cleared.

In order to protect against packet loss, when a signalling change is detected, multiple PW signalling packets are sent. The IWF MUST send a PW signalling packet within *sp\_int* ms. The IWF MUST send PW signalling packets out every *sp\_int* ms until three identical packets are sent. The default value of *sp\_int* is 5 ms.

The IWF SHOULD send a PW signalling frame every *sp\_idle* second in the absence of any signalling change event described above. The default value of *sp\_idle* is 5 seconds. The TSP MUST preserve the ABCD signalling bits at their previous state whenever packets are lost.

## 6.4 Timing aspects

The timing aspects of a TDM-E service are bounded by the existing DOCSIS, ITU and ATIS requirements for DOCSIS transport, and T1 or E1 service. The primary specifications that govern timing and synchronization for a T1 or E1 are the [ITU-T G.823] and [ITU-T G.824] Recommendations.

As required in clauses 6.1.4 and 6.1.5, the T1/E1 interface will be tested for compliance to [ITU-T G.823] or [ITU-T G.824].

### 6.4.1 Direct clocking

A device containing a CU compliant with direct clocking MUST have at least one DTI client interface to a DTI server compliant with [ITU-T J.211] or at least one BITS input for connection to a PRS compliant with [ITU-T G.823] or a PRS compliant with [ITU-T G.824].

A CU embedded in a TEA with direct clocking MUST lock the IWF and TSP to the CU.

#### 6.4.2 Network clock recovery (NCR)

A CU compliant with NCR MUST lock the IWF and TSP to the DOCSIS SYNC message.

A CU compliant with NCR MUST lock the IWF and TSP to the DOCSIS Symbol Clock.

##### 6.4.2.1 Support for NCR

This clause documents the strategy for maintaining long-term interoperability between TE-CMs and TE-CMTSs that use NCR.

###### 6.4.2.1.1 DOCSIS 1.1 and 2.0

Some DOCSIS 1.1 and 2.0 CMTSs may only be able to support the SYNC method. For this reason, all TE-CMs are required to support the SYNC method.

###### 6.4.2.1.2 DOCSIS 3.0

Going forward, the Symbol Clock method is increasingly preferred over the SYNC method. For this reason, all TE-CMs are required to support the Symbol Clock method. TE-CMTSs should support both methods if possible. This Recommendation only requires a TE-CMTS to support one method. [ITU-T J.222.1] requires a CMTS to support both methods.

##### 6.4.2.2 Symbol clock (SC) recovery

One method of locking to the symbol clock (SC) is described as follows. A DTI or BITS source is used to derive the CMTS master clock of 10.24 MHz. The symbol clock is derived from (and locked to) the master clock via an M/N. The TE-CM recovers the 10.24 MHz master clock from the symbol clock via an N/M PLL. The 1.544 MHz T1 clock is recovered from the 10.24 MHz via a 193/1280 PLL. The 2.048 MHz E1 clock is recovered from the 10.24 MHz via a 1/5 PLL. An Ethernet clock (like 25 MHz) could be recovered from 10.24 MHz via a 625/256 PLL. Every clock is in sync.

N/M values – While the values of M and N are configurable at the CMTS, most CMTSs have symbol clocks operating at specific rates. The following information is extracted from [ITU-T J.210].

**Table 6-2 – DOCSIS symbol clocks**

Downstream mode	Nominal specified symbol rate, $f_b$ (MHz)	M/N	Master clock rate, $f_m$ (MHz)	Downstream symbol rate, $f_b'$ (MHz)	Offset from nominal
64QAM ([ITU-T J.83] Annex B)	5.056941	401/812	10.239990	5.056945	0.95 ppm
256QAM ([ITU-T J.83] Annex B)	5.360537	78/149	10.240000	5.360536	0.02 ppm
64QAM or 256 QAM ([ITU-T J.83] Annex A)	6.952	869/1280	10.240000	6.952	0 ppm

- SC TLV – The TE-CM MUST be capable of receiving values for M and N through an 'SC' TLV specific to the TE-CM configuration. If the TE-CM does not receive the SC TLV, it SHOULD use default M/N values from Table 6-2 as selected by the configured QAM modulation.

- S-CDMA override – The TE-CM MUST use the SC-TLV value (or its default value) unless the TE-CM receives an M/N value in the UCD for S-CDMA recovery of 10.24 MHz, in which case the S-CDMA value is to be used. See Table 8-18 of [ITU-T J.122] Channel TLV parameters, Types 12 and 13. The TE-CM MUST report an event if the S-CDMA M/N value overrides the SC TLV.
- Supported values – The TE-CM MUST support the M/N values listed in Table 6-2. The TE-CM SHOULD support all values of M/N as defined in [ITU-T J.122]. The TE-CM MUST report an event if it receives an M/N value it cannot support and revert to the default value from Table 6-2 as selected by the configured QAM modulation.

#### 6.4.2.3 SYNC clock recovery

The existing DOCSIS 1.1, 2.0 and higher describe the DOCSIS SYNC message and the DOCSIS performance requirements. The TE-CMTS using direct clocking uses a frequency reference that meets the [ITU-T G.823] and [ITU-T G.824] requirements to generate the DOCSIS SYNC message and the contained timestamps. The DOCSIS SYNC message not only meets the performance bounds for DOCSIS, it also has long term stability needed for TDM-E. A TE-CM can recover this direct reference clock from the DOCSIS SYNC message timestamps using a PLL. A typical implementation would have a PLL that is disciplined by the DOCSIS SYNC message timestamps to generate an 8 kHz reference for the eTEA that meets the [ITU-T G.823] and [ITU-T G.824] requirements detailed in clause 6.1.4.

#### 6.4.3 Prime differential clock recovery (PDCR)

The IWF SHOULD support prime differential clock recovery.

The principle underlying PDCR is to "encode" the timing attributes of the ingress DS1 service clock using a PRS-traceable reference clock. The peer IWF "decodes" this information using its PRS-traceable reference clock as reference to recreate the service clock for the egress DS1. (In PDCR each TEA has direct clocking or NCR available, but the customer prefers to transport their timing across the network. To do this, the IWF uses the clock from the CU in direct clocking or NCR mode to measure the customer's Tx clock in the TSP. The offset is encoded and sent in an RTP PW. The PW is then received at the second TEA when the offset is added to the clocking from the CU using direct clocking or NCR to time the TSP. Since direct clocking and NCR are traceable to a Stratum 1 reference clock they are equivalent at both TEAs, so the offset measured at one TEA is precisely recreated at the second TEA.)

The PDCR encoding method is described with reference to Figure 6-4 below.

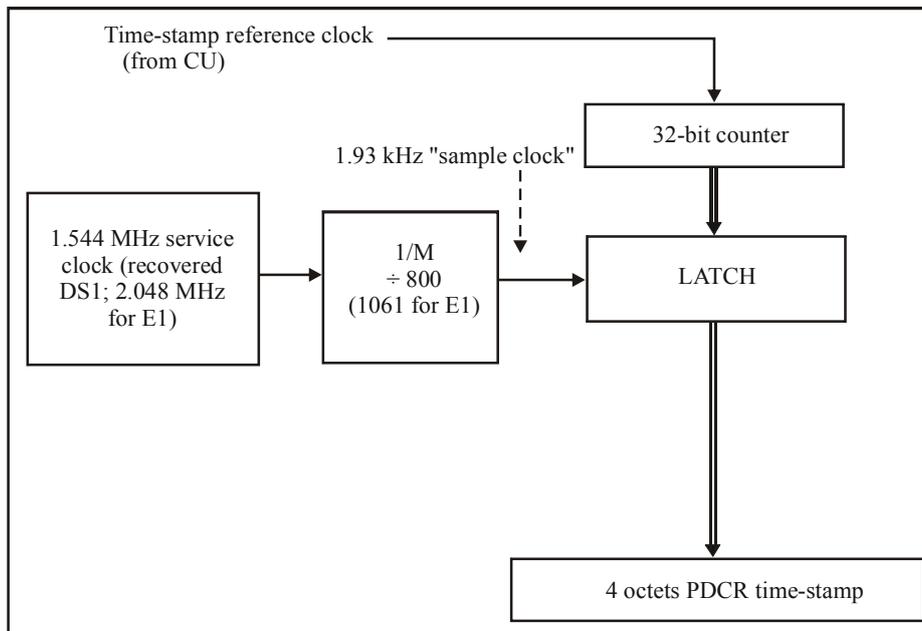
A 32-bit counter forms the base of the PDCR time-stamp (PDCR-TS). The counter operates continually based on the PRS-traceable clock ("Time Stamp reference") with the normal rollover between "all-1s" and "all-0s".

- In PDCR implementations, the IWF MUST use a counter of 32 bits for the PDCR-TS.
- In PDCR implementations, the IWF MUST support RTP.
- In PDCR implementations for T1, the IWF MUST divide the incoming 1.544 MHz "service clock" by  $M = 800$  to generate a 1.93 kHz "sample clock" (PDCR time-stamp-clock).
- In PDCR implementations for E1, the IWF MUST divide the incoming 2.048 MHz clock by  $M = 1061$  to generate a 1.93 kHz "sample clock" (PDCR time-stamp-clock). The generated value will not be precisely 1.93 kHz, but it is close enough.
- The PDCR time-stamp is therefore 4 octets.
- Because two clocks are involved, meta stable states may occur at the output of the latch (if a latch is used). Multiple methods should be used to avoid meta-stable states.

- One method is already specified herein. The occurrence of meta-stable states is minimized because the 1.93 kHz "sample clock" is relatively prime to the 10.24 MHz CM master clock.
- Additional methods SHOULD be devised to avoid meta-stable states.

A PDCR time-stamp is thus generated at the 1.93 kHz rate; and, if, for example, the payload contains 192/193 information octets (i.e., DS1 data), there will be 1 or 2 PDCR time-stamps generated for each DS1-payload unit.

In PDCR implementations, the latest PDCR-time-stamp MUST be sent in the payload.



J.214(07)\_F.6-4

**Figure 6-4 – Description of PDCR "transmit" function**

The PDCR implementation at the receiving IWF uses the packet sequence numbers carried in the CES control word or RTP header to identify missing or out-of-sequence packets.

The PDCR implementation at the receiving IWF uses the incoming PDCR time-stamps and its locally available PRS traceable clock to recover the original service clock.

The PDCR implementation at the receiving IWF can determine the number of PDCR timestamp (service clock) samples using the following formula:

$$\text{Number of samples} = \text{round}\left(\frac{(T_{i+N} - T_i) * SC}{TSRC}\right)$$

Where  $T_i$  and  $T_{i+N}$  are the value of the PDCR time stamps received in packets with sequence number  $i$  and consecutive packet with sequence number  $i+N$  respectively,  $SC$  is the nominal rate of the sample clock (i.e., 1.93 kHz for T1 and 2048/1061 kHz for E1) and  $TSRC$  is the time stamp reference clock (i.e., 10'240 kHz).

For example, assume the DS1 CES packets are sent at a rate of 1 kHz,  $N = 256$ ,  $SC = 1.93$  kHz,  $TSRC = 10'240$  kHz and the offset of the service clock is 10 ppm. For each sample, the timestamp advances by  $10.24\text{M}/(1.93\text{k} + 10 \text{ ppm})$  ticks or equivalently by  $(10.24\text{M}-10 \text{ ppm})/1.93\text{k}$  ticks. The number of samples of the PDCR timestamp between the two packets is either 494 or 495 samples. Therefore, the difference in timestamps would be either  $((10.24\text{M}-10 \text{ ppm})/1.93\text{k}) * 494$  or

$((10.24\text{M}-10\text{ ppm})/1.93\text{k})\cdot 495$ . Therefore, as long as  $10\text{ ppm}\cdot 495$  is much smaller than one, the formula above applies. The DS1 clock offset from the PRS traceable clock is bounded by Telecom standards to be much less than 100 ppm, and therefore the formula above applies as long as N is small enough ( $N < 5000$ ).

When a DS1 is terminated within the DOCSIS network at both ends, both IWFs have access to the 10.24 MHz common time-stamp reference clock. However in situations where the DS1 is transported over the DOCSIS and terminated outside of the DOCSIS network, the external IWF would have access to a PRS traceable clock usually using a dedicated interface to BITS.

In order to support such topologies, the PDCR receiver implementation SHOULD support recovery of DS1 and E1 clocks using PDCR timestamps generated by different master PRS traceable clocks, different sample clocks generated by different divisor M. In particular:

19.44 MHz PRS traceable master clock (DS1 and E1 emulation) with sampling clock as defined by  $M = 800$  for DS1 and  $M = 1061$  for E1 signals.

Selection of the divisor M for other useful frequencies such as 24.704 MHz clock ( $16\cdot 1.544\text{ MHz}$ ) and 32.768 MHz clock ( $16\cdot 2.048\text{ MHz}$ ) is left for future study.

A TEA compliant with PDCR MUST be compliant with clause 6.4.7.

#### **6.4.4 Adaptive clock recovery (ACR)**

A clock recovery unit (CRU) in the IWF compliant with ACR MUST regenerate the clock provided to the TSP based on the packet arrival rate.

The RTP header in ACR can optionally be used in conjunction with the Absolute RTP mode of timestamp generation. Devices operating in the ACR mode work opposite devices operating in the internal or looped timing modes. In the loopback clocking mode, the IWF sets timestamps using the clock received from the incoming TDM circuit. As a consequence, the timestamps are closely correlated with the sequence numbers. All ACR implementations that support RTP must work opposite a TEA operating in this mode.

A CRU compliant with ACR MUST support the use of "absolute mode" timestamps, where the clock used to generate the timestamp is that recovered from the incoming TDM circuit.

A CRU compliant with ACR MUST support the use of "absolute mode" timestamps generated using an 8 kHz clock.

A CRU compliant with ACR MUST support the use of "absolute mode" timestamps generated using clocks with frequencies that are an integer multiples of 8 kHz.

A CRU compliant with ACR MUST be compliant with clause 6.4.7.

#### **6.4.5 TE-CM Ethernet timing service**

A TE-CM MAY support [IEEE 1588] on the Ethernet interface.

- A TE-CM that supports [IEEE 1588] SHOULD use the DOCSIS symbol clock as a Time Source.
- If the TE-CM supports [IEEE 1588], it SHOULD lock the [IEEE 1588] timestamp to the DOCSIS timestamp.

### 6.4.6 TE-CMTS timing requirements

The TE-CMTS symbol (master) clock output MUST be compliant with Figure 6-5, where MTIE (defined in [ITU-T G.810] at observation time (S) is:

$$MTIE(S) = \max_{j=1}^{N-n+1} \left[ \max_{i=j}^{n+j-1} (x_i) - \min_{i=j}^{n+j-1} (x_i) \right]$$

where:

- $\tau$  = sample period
- N = number of samples in the sequence
- $n = \lceil S/\tau \rceil + 1$
- S = observation time
- $x_i$  = time delay sample<sup>1</sup>

MTIE master clock	
Observation time, $\tau$ (s)	MTIE (ns)
$0.1 \leq \tau < 280$	$30 + 10 * \tau + .05 * \tau^2 + 8.75e-5 * \tau^3$
$280 \leq \tau$	$800 + 0.01 * \tau$

**Figure 6-5 – MTIE of CMTS master clock**

The TE-CMTS symbol (master) clock output MUST be compliant with Figure 6-6, where time deviation (TDEV) is defined in [ITU-T G.810].

TDEV master clock	
Observation time, $\tau$ (s)	TDEV (ns)
$0.1 \leq \tau < 200$	6
$200 \leq \tau < 1000$	$0.03 * \tau$
$1000 \leq \tau < 10000$	30

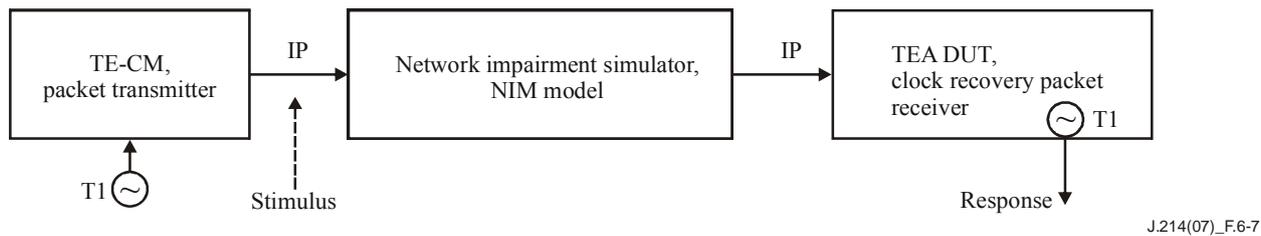
**Figure 6-6 – TDEV of CMTS master clock**

### 6.4.7 Clock recovery performance requirements

Clock recovery methods (ACR and PDCR) are designed to take the TDM clock from at one peer TEA and reproduce that clock at the other peer TEA. This clause outlines the performance requirements of the clock recovery methods. Further, since the performance of these methods may be affected by the characteristics of the intervening IP network, this clause also defines a model of the intervening network; defines the characteristics of that model; and states the performance

<sup>1</sup> Credit [ITU-T J.211].

requirements in the context of the model characteristics. Figure 6-7 shows the model system within which the clock recovery system must meet performance requirements.



**Figure 6-7 – Clock recovery – Performance test environment**

### 6.4.7.1 Definitions

- Stimulus – The stimulus to the performance test environment consists of IP packets containing clock recovery data. These may be generated directly by a PC, or by a 'golden' TE-CM driven by a top quality T1 clock. The IP packets entering the TEA DUT may come over an RF port (associated with a TE-CM), or over the Ethernet port in a special test mode.
- NIM – The stimulus packet stream will be delayed (and otherwise altered for test purposes) by the network impairment simulator operating according to one of the NIM models (see clauses 6.4.7.2 and 6.4.7.3). The NIM simulator will be concatenated with an unloaded DOCSIS system to mimic DOCSIS effects like the upstream grant process where the T1 clock may beat against the DOCSIS clock.
- DUT – The TEA device under test must reproduce the T1/E1 clock within the quality requirements of a T1/E1 clock and within the performance requirements of clause 6.1.4.

### 6.4.7.2 Network impairment model (NIM1)

Two different network impairment models are defined in this Recommendation. This clause defines NIM1; NIM2 is defined in clause 6.4.7.3. Either NIM may be used to demonstrate compliance as outlined in clause 6.4.7.4.1. The NIM chosen is then referred to herein only as "NIM".

This network impairment model (NIM1) will be the Gilbert-Elliott model as outlined in Appendix I of [ITU-T G.1020] and in [TIA NIM]. NIM1 will be configured according to clause 6.4.7.2.1 as configured according to the NIM1 constants (clause 6.4.7.2.2). It is an assumption that any TE-CM/TS involved in the transport of IP packets in Figure 6-7 does not add significant jitter, packet loss, or packet reordering to the IP stream when compared to the NIM effects. It is not expected that the NIM will sufficiently replicate the characteristics (not yet identified) of the real network. It is only expected that the NIM will sufficiently stress a clock recovery circuit for testing purposes.

#### 6.4.7.2.1 NIM1 state machine

The Gilbert-Elliott NIM1 can be modelled per [ITU-T G.1020]. The Gilbert-Elliott model behaves according to a state machine with two states:

- State 0 is a low loss state with loss probability PL(0) and a state transition probability PT0.
- State 1 is a lossy state with loss probability PL(1) and a state transition probability PT1. It should be noted that  $PL(1) \gg PL(0)$ .

Packet loss, state transition, and packet delay are driven by a Markov process. There is an option allowing (or disallowing) packet loss. The following is a pseudocode implementation of the state machine. It contains initialization, a loop that computes the delay for each packet, and a final reporting section.

```

/* Version NIM4.C */
#include <c:\Progra~1\turboc\total\stdio.h>
  
```

```

#include <c:\Progra~1\turboc\total\time.h>

/* input constants (to be filled in from the NIM constants table) */
float PL[2] = {.01, .01};      /* the prob of packet loss in @ state */
float PT[2] = {.04, .5};      /* the prob of state transition in @ state */
float PI = .2;                 /* the prob of delay impulse */
float HI = 60.;                /* the delay driver, impulse height */
float FL = 20.;                /* the delay impulse smoothing filter */
int AllowPacketLoss = 1;       /* is packet loss allowed ? */
int AllowPacketReorder = 0;    /* is packet reordering allowed ? */

/* program constants */
int numPackets = 1000;
float lost = 32000.;           /* an encoding delay signifying 'lost' */
int states = 2;                /* a two state Gilbert-Elliott model */
float packetTime = 1.;        /* in ms (estimated), for packet ordering */

/* variables */
int state;                     /* 0 for low loss, 1 for high loss */
float d;                       /* the delay the packet will undergo */
float ds;                       /* the accumulated & filtered delay state */
float impulse;                 /* the raw height of a delay impulse */
float lastd;                   /* tracks prev delay, reordering needed? */

int numDiscarded = 0;          /* number of discarded packets */
int numNotDiscarded = 0;       /* number not discarded */
float cumDelay = 0.;           /* Constants to compute variance of delay */
float cumDelaySquare = 0.;
float fnum, fdenom, Variance;
int PDiscarded, PNotDiscarded;

FILE *stream;
int i, s, packet;

main() {

    stream = fopen("myfile.txt", "w");
    /* The following code is executed before packets are processed */
    printf(" PL[0] %4.3f\n", PL[0]);
    printf(" PL[1] %4.3f\n", PL[1]);
    printf(" PT[0] %4.3f\n", PT[0]);
    printf(" PT[1] %4.3f\n", PT[1]);
    printf(" PI %4.3f\n", PI);
    printf(" HI %4.3f\n", HI);
    printf(" FL %4.3f\n", FL);

    /* initialization */
    srand( (unsigned int) time( NULL ) ); /* seed the random generator */
    ds = d = lastd = 0; /* initialize the delay state */
    /* ----- */
    /* after initialization, for every packet... */
    for (packet = 0; packet < numPackets; packet++) {
        if (rand()/32768. < PI) { impulse = HI; } /* determine delay */
        else { impulse = 0.; }
        ds = (ds*(FL - 1) + impulse) / FL; /* filter the raw delay */
        d = ds; /* delay */
        if ((AllowPacketLoss == 1) && (rand()/32768. < PL[state])) {d += lost;}
        if (rand()/32768. < PT[state]) {state = ((state + 1) & 1);}
        if ((AllowPacketReorder == 0) && (d < (lastd - packetTime))) {
            d = lastd - packetTime;
            printf("d \t%4.1f\tlastd \t%4.1f \t reorder ", d, lastd);
        }
        else {
            printf("d \t%4.1f\tlastd \t%4.1f \tno reorder ", d, lastd);
        }
        if (d < lost) {lastd = d;} /* not discarded */
        else {
            if (lastd >= packetTime) {lastd -= packetTime;} /* discarded */
        }
        if ((AllowPacketLoss == 1) && (d >= lost)) {
            /* {discardPacket();} */
        }
    }
}

```

```

        printf("Discarded \n");
        numDiscarded++;
    }
    else {
        /* ScheduleToForwardPacketAfterDelay(d); */
        printf("ForwardAfterDelay %8.1f\n",d);
        numNotDiscarded++;
        cumDelay += d;
        cumDelaySquare += d*d;
    }
    fprintf(stream,"%4.1f\n",d);
} /* -----*/
printf("\nDone\n");
fclose(stream);
PDiscarded = (100 * (double)numDiscarded)/numPackets;
PNotDiscarded = (100 * (double)numNotDiscarded)/numPackets;
printf("Number Discarded\t= %6i %3i Percent \n"
        ,numDiscarded, PDiscarded);
printf("Number Not Discarded\t= %6i %3i Percent \n"
        ,numNotDiscarded, PNotDiscarded);
printf("Average Delay\t=\t %4.1f\n", (cumDelay/numNotDiscarded));

/* StdDev is sqrt( (n* sum(x^2) - sum(x)^2) / ((n(n-1)) ) */
fnum = ((float)numNotDiscarded * cumDelaySquare) - (cumDelay * cumDelay);
fdenom = (float)numNotDiscarded * ((float)numNotDiscarded - 1);
Variance = (fnum/fdenom);
printf("Delay Variance\t=\t%5.1f\n", Variance);
return;
} /* end */

```

NOTE 1 – The FL is used to filter and smooth the delay impulses as they become delays. The term  $(ds*(FL - 1) + impulse) / FL$  shows that the entries in FL should be picked to be larger than 1. Variable ds provides delay memory between packets and decays because of the  $-1$ . If FL is set high, it will decay (in the absence of impulses) slowly. If FL is set to 2, it will decay rapidly. FL should be determined first. Then HI can be set to determine the amplitude of the impulses and delays. Note that impulses are divided by FL. This is the reason to set FL first, then determine the size of HI needed to drive the impulses.

NOTE 2 – The function ScheduleToForwardPacketAfterDelay(d) engenders the need for multi-threading in the software. No pseudo code is included for this function (or for discardPacket()).

#### 6.4.7.2.2 NIM1 constants

The NIM1 constants configure NIM1 according to the Gilbert-Elliott model as outlined in [ITU-T G.1020]. The specific enumeration of the constants in Table 6-3 serves to define and limit the possible variations of the Gilbert-Elliott model.

The following constants pertain to the performance requirements in clause 6.4.7.4.1. To further simulate network characteristics that vary during the day (ToD), two separate columns of NIM1 constant values are specified below in Table 6-3. Each NIM1 constant in the pseudo code of clause 6.4.7.2.1 may be faded from one ToD value to the other (and back) on a time scale simulating real ToD variations of the network.

**Table 6-3 – Network impairment constants (NIM1)**

Item	Description	Value ToD1	Value ToD2	Value ToD3 (Excessive packet loss)	Format	Model variable name
Probability of packet loss	The probability of packet loss in each state. (The lossy state is second)	.01/.01	.01/.01	.01/.99	float	PL[state]
Probability of state transition	The probability of state transition in each state	.04/.5	.04/.5	.5/.001	float	PT[state]
Probability of delay	The probability of impulse	.2	.2	.2	float	PI
Amount of delay	The delay impulse height	60	60	60	float	HI
Delay smoothing filter	The smoothing factor for delays	20	6	6	float	FL
Packet Loss	Determines whether the model is allowed to lose packets	1 (true)	1 (true)	1 (true)	int	AllowPacketLoss

**6.4.7.3 Network impairment model (NIM2)**

NIM2 is defined according to [ITU-T G.1050] and configured according to the constants in Table 6-4. This set of constants should generate a [ITU-T G.1050] NIM with the characteristics outlined in Table 6-5. The normative requirements are defined by Table 6-4 whereas Table 6-5 is informative only.

**Table 6-4 – Network impairment constants (NIM2)**

<b>G.1050 constant</b>	<b>Value</b>
Service Profile	B
Test Case	127
LAN A Occupancy*	0
Access A Occupancy*	0
MTU A*	512 bytes
Route flap interval*	None
Route flap duration*	0
Delay (regional)*	0
Delay (intercontinental)*	0
Link fail interval*	None
Link fail duration*	0
Packet loss*	0
Reordered packets*	0
Access B Occupancy*	0
MTU B*	512 bytes
LAN B Occupancy*	0
* This minimizes network effects that might be generated by the [ITU-T G.1050] model, leaving only generated delays and packet losses.	

**Table 6-5 – NIM2 characteristics**

<b>NIM2 characteristics</b>	<b>Value</b>
Latency	16 ms
Jitter	3 ms
Loss	0.1%
Average Sequential Loss	3.3
Max Sequential Loss	7
Sequential Loss Rate	0.2
OoS	0.003%
Discards	0%

#### **6.4.7.4 Performance requirements**

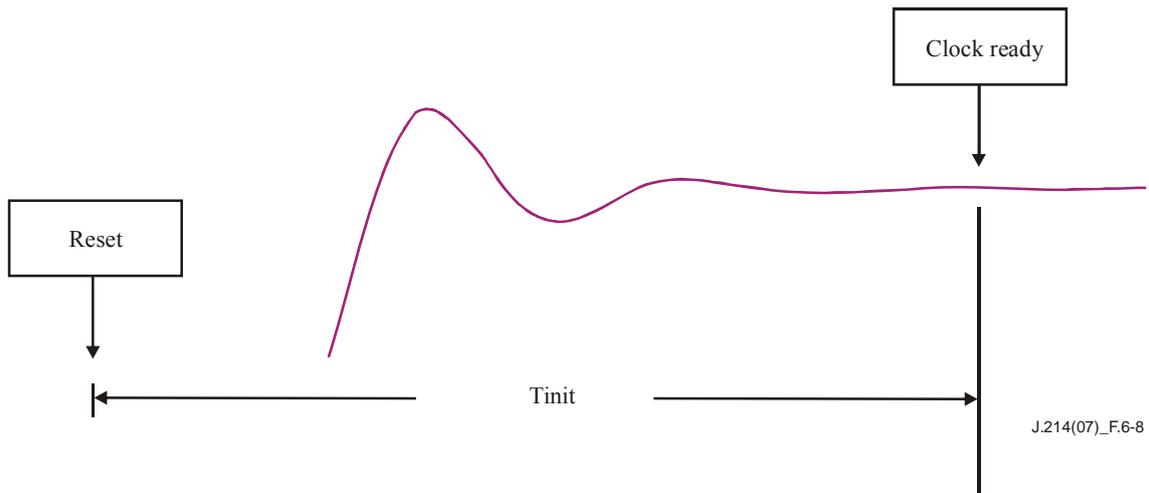
This clause specifies that clock recovery should function in the presence of network impairments like jitter and packet loss. It should be noted that PDCR should not be susceptible to network impairments like delay or jitter because they are subtracted out.

##### **6.4.7.4.1 Specific requirements**

The following requirements only apply to the clock recovery unit within the IWF when operating in PDCR or ACR mode implemented according to clauses 6.4.3 and 6.4.4. Figures 6-8 through 6-10 are informative only, serving only to illustrate measurements. The tests which follow, and the informative figures, are based on the assumption that the clock recovery mechanism can be modelled as a linear, second order control loop. In that context, there are no requirements on ringing

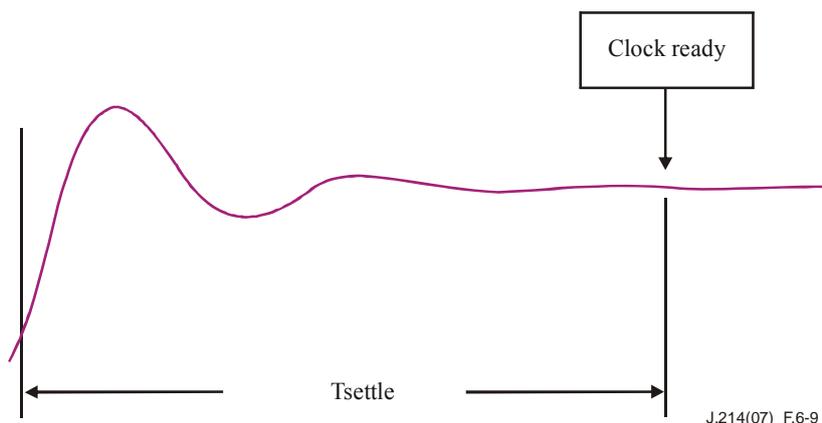
frequency, overshoot, or damping; the control loop may be over or under damped and the measurement techniques still apply.

- An IWF MUST meet the requirements in this clause in the face of network impairments described in clause 6.4.7 and generated by either NIM1 specified in clause 6.4.7.2 or NIM2 specified in clause 6.4.7.3.
- Table 6-6 is part of the following requirements; it contains the constants to which they refer. A TEA MUST have a configurable test mode so that the PW packets may be looped back directly to the TEA (as a replacement for packets received at the TE-CM RF port).
- Initialization – A CRU MUST acquire an existing stream of pseudo wire IP packets and lock the TDM output clock within  $T_{init}$  seconds during the initialization of the TEA.



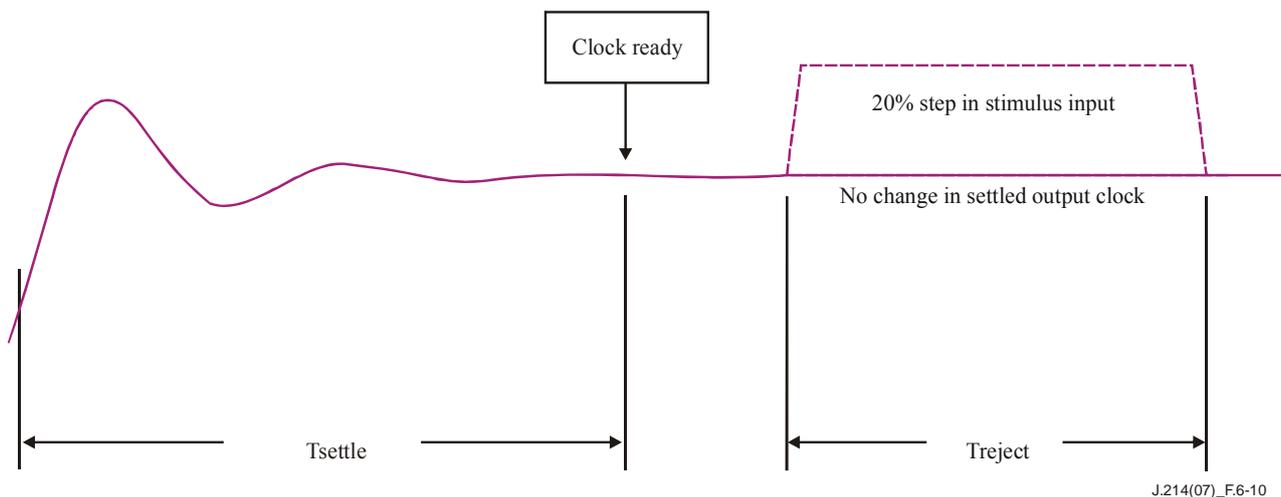
**Figure 6-8 – Clock recovery initialization time**

- Step response settling time – A CRU within the IWF MUST be capable of the following clock recovery sequence: settle for  $T_{init}$  seconds to a stable locked frequency; accept a 20% step up from nominal of the stimulus frequency; and re-settle its TDM clock output to acceptable TDM clock requirements in clause 6.1.4 within  $T_{settle}$  seconds after the step input.



**Figure 6-9 – Clock recovery settling time**

- Rejection of impulse noise – A CRU MUST be capable of rejecting a  $T_{reject}$  second long, 20% (from nominal) amplitude deviation of stimulus frequency (after it has settled) without disturbing its clock output.



**Figure 6-10 – Rejecting impulse noise**

- Capture range – A CRU MUST successfully acquire, recover, and track TDM input stimulus frequencies within  $T_{capture}$  Hz of the nominal TDM output clock frequency, where  $T_{capture}$  may have different values for T1 and E1.
- Rejection of temporary change in average delay – A CRU MUST be capable of rejecting a  $T_{vary}$  second long, 20% (from nominal) change of average delay without disturbing its clock output. This temporary change of average delay may be created by the NIM model.
- Rejection of permanent change in average delay – A CRU MUST be capable of rejecting a permanent 20% (from nominal) change of average delay without disturbing its clock output. This change of average delay may be created by the NIM model.
- Rejection of excessive packet loss condition – A CRU MUST be capable of keeping the clock output stable throughout a single excessive packet loss rate condition of duration  $T_{loss}$  seconds. This excessive packet loss rate condition may be created by the NIM model using the ToD3 constants for  $T_{loss}$  seconds.

#### 6.4.7.4.2 Performance requirement constants

The following constants pertain to the performance requirements in clause 6.4.7.4.1.

**Table 6-6 – Clock recovery performance constants**

Item	Description	ACR value	PDCR value	Units
Tinit	The time between initialization (reset) and clock recovery.	700	700	Seconds
Tsettle	The settling time of the output frequency.	900	900	Seconds
Treject	The length of stimulus noise impulse that must be rejected.	1	1	Seconds
Tcapture	The maximum deviation of input frequency over which recovery functions properly.	$\pm 50$ (T1) $\pm 60$ (E1)	$\pm 50$ (T1) $\pm 60$ (E1)	Hz
Tvary	The length of a temporary change in average delay.	1	1	Seconds
Tloss	The duration of an excessive packet loss condition.	1	1	Seconds

## 6.5 Defects and monitoring

### 6.5.1 Alarm states and alarms

- *LoPS (loss of packet state)*

An IWF MUST cause a PW to enter the LoPS state when more than the configurable percentage of packets are lost over a configurable packet loss time window or when a configurable number of consecutive packets are missing.

If a PW is in the LoPS state, the IWF MUST set the R bit in transmitted packets.

If a PW is in the LoPS state, the IWF MUST cause the PW to exit the state when a configurable number of consecutive packets are played out of the jitter buffer with no errors or when a complete jitter buffer has been played out with no missing packets.

- *Excessive packet loss alarm (EPL)*

If the IWF stays in the LoPS state for more than 2.5 seconds, it MUST set the EPL Alarm.

The IWF MUST clear the EPL alarm after 10 seconds of no errors.

- *Errored seconds*

The IWF MUST detect and count both errored seconds, severely errored seconds, and unavailable seconds for each PW.

The IWF MUST detect an error defect whenever a filler data is played out of the jitter buffer. For structure agnostic operation, a filler packet is considered as one error defect.

The IWF MUST count an errored second as any second in which one or more error defect is detected.

The IWF MUST count a severely errored second as a second in which the number of error defects is equal to or greater than the configurable severely errored second threshold number of error defects.

The IWF MUST count an unavailable second when ten consecutive seconds qualify as severely errored seconds, and it continues to be unavailable until ten consecutive seconds occur that do not qualify as severely errored seconds.

- *Fill values*

For missing data or packets, or when in the LoPS state, the IWF MUST fill transmitted TDM signals with the configured fill value.

In structure-agnostic mode, the IWF MUST use the configured fill value. The IWF MAY use the AIS code as the configured fill value in structure agnostic mode.

In structure-aware mode, the IWF MUST use the configured fill value for the applicable timeslots and maintain transport framing. The IWF MAY use the AIS code as the configured fill value in structure-aware mode.

### 6.5.2 Packet reordering

The IWF MUST detect mis-ordered PW packets. The IWF SHOULD reorder mis-ordered PW packets. If the IWF cannot reorder a mis-ordered packet, it MUST discard the packet. All mis-ordered packets that are not reordered MUST be counted.

### 6.5.3 Packet loss concealment

When occasional packet loss occurs on a PW, the IWF MUST maintain framing and fill the TDM stream with a configurable fill octet; default all ones.

In extended superframe (ESF) mode on T1, the IWF MUST set the ESF data link signalling bits to 1's in a PW packet loss condition.

#### **6.5.4 Packet loss statistics**

The IWF MUST maintain a MIB for each PW with 24 hours/96 intervals of 15-minute packet loss statistics. In each 15-minute interval, the IWF MUST update MIB objects maintaining a count of lost packets and a count of seconds where the SPLA alarm condition has been declared.

#### **6.5.5 Stray packets**

An IWF MUST discard stray (non-PW) packets. An IWF MUST NOT alter its packet loss statistics based on discarding stray packets.

#### **6.5.6 Packet activity during provisioning**

An IWF MUST output an 'all ones' pattern as its TDM flow to the TSP while it is being set up (before it is ready to re-transmit received data) and while it is being torn down (after it no longer can retransmit). These time periods may be defined by the fill state of the IWF buffer for a given PW.

### **6.6 Security**

PW traffic between IWF peers may cross networks where security is an issue. While latency is an issue with any further processing, [IETF RFC 3711] seems to be a very efficient and tight security mechanism for this application.

- The IWF MAY implement [IETF RFC 3711] security for end-to-end security between peer IWFs in a service.
- If an IWF implements [IETF RFC 3711], the IWF MUST be able to optionally disable it.
- The IWF may use other security protocols like IPSec.
- The IWF MUST NOT use security protocols that cannot be disabled.
- Key exchange is not fully specified in [IETF RFC 3711] and is out of scope herein.

### **6.7 Provisioning and service initiation**

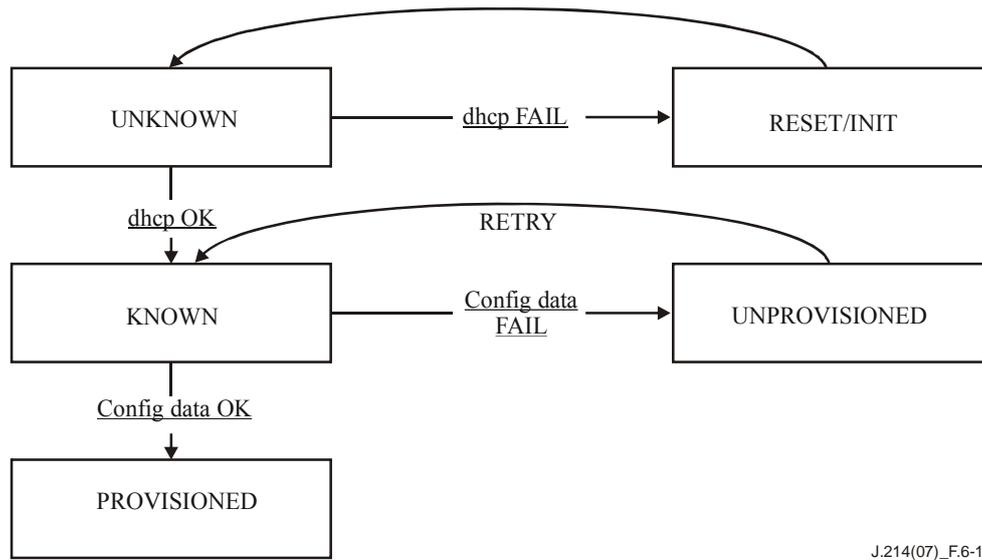
#### **6.7.1 Provisioning of embedded TDM emulation adapter (eTEA)**

This clause describes the embedded TDM emulation adapter (eTEA), its attributes and its configuration.

##### **6.7.1.1 eTEA device provisioning**

The eTEA is defined as a eSAFE device as specified by [ITU-T J.126]. Device provisioning is the process by which the eTEA device is configured to support TDM Emulation service. The device provisioning involves the eTEA obtaining its IP configuration required for basic network connectivity, announcing itself to the network, and downloading its configuration data from its provisioning server. A basic content integrity verification check MUST be conducted on the configuration file by the eTEA as defined in clause 6.7.1.9.4.

Figure 6-11 illustrates the eTEA provisioning process.



**Figure 6-11 – Device states and state transitions for basic flow provisioning**

#### 6.7.1.2 eTEA endpoint provisioning

After an eTEA is successfully provisioned, it has to establish transport connectivity with its peer TEA or eTEA to transition to an operational state and ready to carry traffic. Such end-to-end provisioning and monitoring protocol is at the discretion of the operators, and [ID VCCV] may be used for such purpose.

#### 6.7.1.3 Security

An eTEA is provisioned through a DOCSIS-like provisioning flow and should use standard DOCSIS security methods.

#### 6.7.1.4 Backoff, retries and timeouts

The backoff mechanisms help the network to throttle device registration during a typical or multiple client registration condition when requests are not serviced within the protocol specified timeout values. This Recommendation does not delve into the details of these requirements but outlines the following as a general guideline:

- The recommendation for the throttling of registration may be based on [ITU-T J.112] CM registration.
- The eTEA must follow DHCP specifications for the timeout and retry mechanisms.
- The eTEA must use an adaptive timeout for TFTP as specified in [ITU-T J.112].

#### 6.7.1.5 Power-on initialization flow

The eTEA first obtains its IP address along with other information including its config file name and address, and then download the eTEA config file from the TFTP server. This process is illustrated in Figure 6-12 and described in more detail in Table 6-7.

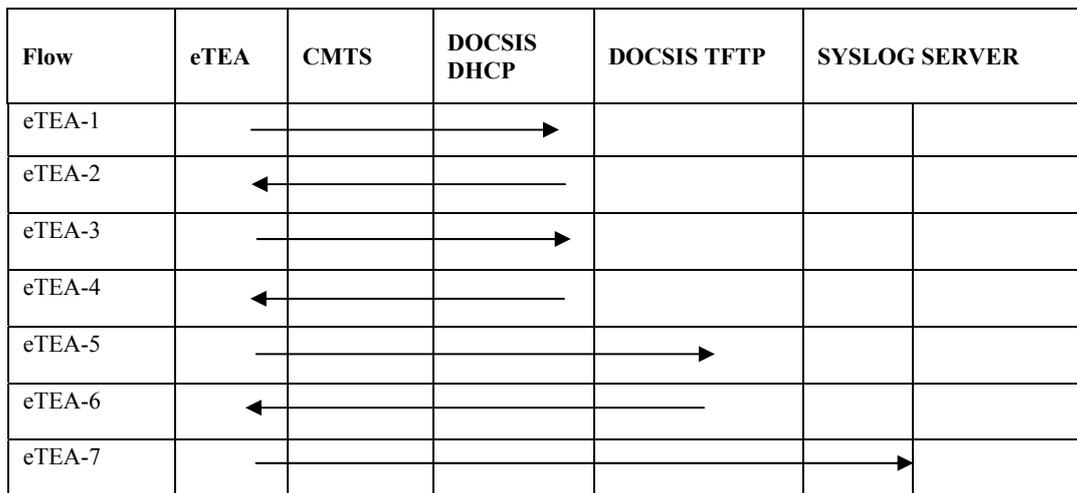


Figure 6-12 – eTEA power-on initialization flow

Table 6-7 – eTEA initialization flow

Step	eTEA initialization description	Normal flow sequencing	Must proceed to here if this step fails
eTEA-1	<p>DHCP Broadcast DISCOVER</p> <p>The eTEA MUST send a broadcast DHCP DISCOVER message. This message MUST include option code 60 (vendor specific option) in the format "bsod-te1.0:xxxxxx". The eTEA MUST include the DHCP option code 43 in the DHCP DISCOVER message as defined in clause 6.7.1.8.3. The eTEA MUST request in DHCP option 55 the following: option 1 (subnet mask). The eTEA MAY request in DHCP option 55 the following: option 7 (syslog server).</p>	eTEA-1 MUST NOT occur before CM registration completion.	eTEA-1
eTEA-2	<p>DHCP OFFER</p> <p>The eTEA MUST only accept a valid DHCP OFFER message. A valid DHCP OFFER MUST include option 1, the configuration file location in the "siaddr" field and the configuration file name in the "file" field.</p>	eTEA-2 MUST occur after eTEA-1 completion.	eTEA-1
eTEA-3	<p>DHCP broadcast REQUEST</p> <p>The eTEA MUST send a DHCP REQUEST after selecting a valid DHCP OFFER.</p>	eTEA-3 MUST occur after eTEA-2 completion.	eTEA-1

**Table 6-7 – eTEA initialization flow**

Step	eTEA initialization description	Normal flow sequencing	Must proceed to here if this step fails
eTEA-4	<p>DHCP ACK</p> <p>The DHCP server sends a DHCP ACK message to the eTEA. The DHCP ACK message MUST include all options and sub-options which had been sent in eTEA-2 (DHCP OFFER). If the option and sub-option values of this DHCP ACK differ with the preceding DHCP OFFER (eTEA-2), the option and sub-option values of this DHCP ACK MUST be treated as authoritative (per [IETF RFC 2131]).</p> <p>If the DHCP ACK is not valid as per the criteria established in eTEA-2, the eTEA MUST fail this step.</p>	eTEA-4 MUST occur after eTEA-3 completion.	eTEA-1
eTEA-5	<p>TFTP Configuration File Request</p> <p>The eTEA MUST perform a TFTP protocol exchange to download its configuration file. The 'siaddr' and 'file' fields of the DHCP ACK are used to locate the configuration file. Specific details of the TFTP protocol can be found in [IETF RFC 1350].</p>	eTEA-5 MUST occur after eTEA-4 completion.	eTEA-1
eTEA-6	<p>TFTP Configuration File Response</p> <p>The TFTP server MUST send the requested configuration file to the eTEA. Specific details of the TFTP protocol can be found in [IETF RFC 1350].</p> <p>The number of TFTP retries is governed per [ITU-T J.122].</p>	eTEA-6 MUST occur after eTEA-5 completion.	<p>If the configuration file download failed per TFTP protocol, go to eTEA-1.</p> <p>If the configuration file itself is in error, log the error and end the process.</p>
eTEA-7	<p>SYSLOG NOTIFICATION</p> <p>If the Syslog server IP address has been received, the eTEA MUST send a provisioning complete notification. This should include the PASS/FAIL result of the provisioning operation.</p>	eTEA-7 MUST occur after the completion of eTEA-6.	

The eTEA configuration file should contain all the necessary parameters to provision every entity including the PW, TSP Ethernet and CU, within the eTEA and the eTEA itself. These entities inside eTEA may be enabled or disabled individually via SNMP.

#### 6.7.1.6 Behaviour during a disconnected state

Changes in the state of the connection can be reported by each eTEA.

- The eTEA MUST define 'starting' and 'stopping' for signalling and data trap activity over a configurable interval with a default of 30 seconds.
- The eTEA MUST limit the combined TDM signalling and TDM data traps to a configurable number in any one hour with a default setting of 24. The eTEA should not control the rate until the limit is reached; the total sent may be reset each hour.

- TDM signalling traps:
  - When a TEA senses that its peer has stopped sending TDM signalling packets, the TEA SHOULD send a TDM signal link down trap.
  - When a TEA senses that its peer has started sending TDM signalling packets, the TEA SHOULD send a TDM signal link up trap.
- TDM data traps:
  - When a TEA senses that its peer has stopped sending TDM data packets, the TEA SHOULD send a TDM data link down trap.
  - When a TEA senses that its peer has started sending TDM data packets, the TEA SHOULD send a TDM data link up trap.

#### **6.7.1.7 Temporary signal loss**

If the eCM or DOCSIS reset for any reason, the eTEA MUST reset and reinitialize, this will result in the TDM service disruption till the configuration parameters and the operational status of the eTEA is restored.

#### **6.7.1.8 DHCP options**

##### **6.7.1.8.1 Standard DHCP options**

DHCP options 1, 2, 3, 4 and 7 are defined in [IETF RFC 2132].

Option code 1 (Subnet Mask).

Option code 2 (Time Offset).

Option code 3 (Router Option/Default Gateway).

Option code 4 (Time Server Option).

Option code 7 (Log Server Option).

##### **6.7.1.8.2 DHCP Option 60: Vendor client identifier**

Option code 60 contains a string identifying capabilities of the eTEA. The eTEA MUST send the following ASCII Coded String in DHCP Option code 60: "bsod-te1.0:zzzzzz". Where zzzzzz MUST be an ASCII representation of the hexadecimal encoding of the eTEA TLV Encoded Capabilities, as defined in clause 6.7.1.11.

##### **6.7.1.8.3 DHCP Option 43**

The eTEA MUST send the DHCP Option 43 in the DHCP DISCOVER and DHCP REQUEST.

DHCP Option 43 contains the number of sub-options defined to provide the eTEA device specific information to the back-office systems. The DHCP option 43 sub-options 1 through 10 and 31 are specified by TDM-E. The TDM-E DHCP option 43 sub-options MUST be present in the format of "Encapsulated vendor-specific extensions" ([IETF RFC 2131]).

Table 6-8 contains the sub-options of the DHCP Option-43, which the eTEA MUST use. The eTEA MUST send all required sub-options listed in the table below unless explicitly stated otherwise. If the total number of octets in all DHCP option 43 sub-options exceeds 255 octets, the eTEA MUST follow [IETF RFC 3396] to split the option into multiple smaller options.

**Table 6-8 – DHCP Option 43 syntax**

eTEA DHCP Option 43	Required/Not used	Value	Description
Sub-option 1	Not Used		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. The DHCP option 43 sub-option 1 MUST NOT be used by the eTEA.
Sub-option 2	Required	"E TEA"	The sub-option 2 contains the device type of the component making the DHCP request. The eTEA MUST send the DHCP option 43 sub-option 2. For TDM-E, this is:  "E TEA"= embedded TEA.
Sub-option 3	Not Used		The sub-option 3 contains a colon separated list of all components in the eDOCSIS device. It is used by the eDOCSIS eCM device. The DHCP option 43 sub-option 3 MUST NOT be sent by the eTEA.
Sub-option 4	Required	<device serial number>	The sub-option 4 contains the device serial number represented as an ASCII string. The eTEA MUST send the DHCP option 43 sub-option 4. The DHCP option 43 sub-option 4 value MUST be identical to the value of the PWDevSerialNumber MIB Object.
Sub-option 5	Required	<Hardware version>	The sub-option 5 contains the hardware version number represented as an ASCII string.
Sub-option 6	Required	<Software version>	The sub-option 6 contains the software version number represented as an ASCII string. The eTEA MUST send the DHCP option 43 sub-option 6. The DHCP option 43 sub-option 6 value MUST be identical to the value of the PWDevSwCurrentVers MIB object.
Sub-option 7	Required	<Boot ROM version>	The sub-option 7 contains the Boot ROM Version represented as an ASCII string. The eTEA MUST send the DHCP option 43 sub-option 7. The DHCP option 43 sub-option 7 value MUST be identical to the <Boot ROM version> field in MIB II object sysDescr.
Sub-option 8	Required	<OUI>	The sub-option 8 contains the organizational unique identifier (OUI) represented as a hexadecimal-encoded 3-byte octet string. The eTEA MUST send the DHCP option 43 sub-option 8. If omitted, the DHCP Server SHOULD use the eTEA MAC address as the eTEA OUI.
Sub-option 9	Required	<Model number>	The sub-option 9 contains the eTEA device model number represented as an ASCII string. The eTEA MUST send the DHCP option 43 sub-option 9. The DHCP option 43 sub-option 9 value MUST be identical to <Model Number> field in the MIB-II object sysDescr.

**Table 6-8 – DHCP Option 43 syntax**

eTEA DHCP Option 43	Required/Not used	Value	Description
Sub-option 10	Required	<Vendor name>	The sub-option 10 contains the Vendor Name represented as an ASCII string. The eTEA MUST send the DHCP option 43 sub-option 10. The DHCP option 43 sub-option 10 value MUST be identical to <Vendor Name> field in the MIB-II object sysDescr.
Sub-option 31	Required	<eTEA MAC Address>	The sub-option 31 contains the eTEA MAC Address encoded as a 6-byte octet string. The eTEA MUST send the DHCP option 43 sub-option 31. The DHCP option 43 sub-option 31 value MUST be identical to the content of the MacAddress MIB object.

### 6.7.1.9 eTEA configuration file

The eTEA configuration file follows the CM configuration format specified in [ITU-T J.122].

#### 6.7.1.9.1 eTEA binary configuration file format

The eTEA-specific configuration data MUST be contained in a file which is downloaded to the eTEA via TFTP. This is a binary file in the same format defined for DHCP vendor extension data [IETF RFC 2132].

The configuration file MUST consist of a number of configuration settings, each of the form "Type Length Value (TLV)". Definitions of these terms are provided in Table 6-9.

**Table 6-9 – TLV definitions**

<b>Type</b>	A single-octet identifier which defines the parameter.
<b>Length</b>	A single-octet field specifying the length of the value field (not including Type and Length fields).
<b>Value</b>	A set of octets length long containing the specific value for the parameter.

The configuration settings MUST follow each other directly in the file, which is a stream of octets (no record markers).

#### 6.7.1.9.2 eTEA configuration file settings

An eTEA MUST be capable of processing all standard configuration settings. An eTEA MUST ignore any configuration setting present in the configuration file which it cannot interpret.

An eTEA configuration file MUST include:

- the eTEA message integrity check (MIC) (Type 53).
- the end of data marker (Type 255).

An eTEA configuration file MAY include the other TLVs defined in clause 6.7.1.10.

Authentication of the provisioning information is provided by a message integrity check (MIC), eTEA MIC. It is a digest which ensures that the data sent from the provisioning server were not modified en route. This is NOT an authenticated digest (it does not include any shared secret).

### 6.7.1.9.3 Configuration file creation

The sequence of operations required to create the configuration file is as follows:

- 1) Create the type/length/value entries for all the parameters required by the eTEA.
- 2) Calculate the eTEA message integrity check (MIC) configuration setting as defined in clause 6.7.1.9.4 and add to the file following the last parameter using code and length values defined for this field.
- 3) Add the end of data marker.

#### 6.7.1.9.4 eTEA MIC calculation

The eTEA message integrity check configuration setting **MUST** be calculated by performing an MD5 digest over the bytes of the configuration setting fields. It is calculated over the bytes of these settings as they appear in the TFTP image, without regard to TLV ordering or contents. There is one exception to this disregard of the contents of the TFTP image:

The bytes of the eTEA MIC TLV itself are omitted from the calculation. This includes the type, length, and value fields.

On receipt of a configuration file, the eTEA **MUST** recompute the digest and compare it to the eTEA MIC configuration setting in the file. If the digests do not match, then the configuration file **MUST** be discarded.

#### 6.7.1.10 eTEA provisionable attributes

The eTEA provisionable attributes are defined as TLVs in the following subclause.

##### 6.7.1.10.1 Configuration file format requirements

The configuration settings **MUST** follow each other directly in the file, which is a stream of octets. The eTEA **MUST** be capable of properly receiving and processing the configuration file.

An eTEA configuration file contains the parameters to configure in the TDM interface using TLV 11 on the standard DS1 and DS0 MIBs, configuration of the cable modem via TLV8, and configuration of the PWs using TLV 9 and 10.

An eTEA configuration file **MUST** contain the eTEA message integrity check (MIC) (Type 53) and the end of data marker (Type 255).

#### eTEA Symbol Clock

This object configures the parameters for dividing down the reference clock and contains the M, N values.

Type	Length	Value
8	4	m,n

This TLV contains the values for the M/N PLL.

#### eTEA IWF configuration encoding

This object configures the interworking function of the eTEA. It consists of a number of encapsulated type/length/value fields. The encapsulated fields define the individual parameters for the IWF being configured. Within the IWF, each supported PW is configured separately as shown below. Parameters that are typically configured identically for many PWs are grouped together in TLV type 10.

Type	Length	Value
9	n	

### eTEA PW Index setting

This object configures the index number of the PW. The settings for each individual PW are preceded by explicitly setting the index number of the PW.

Type	Length	Value
9.1	2	<PW Index>

### eTEA PW Emulation Type

This object configures the emulation type of the PW. The default value is t1Satop(17).

Type	Length	Value
9.2	1	<e1Satop(17), t1Satop(18), basicCesPsn(21), tdmCasCesPsn(23)>

### eTEA PW Peer Address

This object configures the IPv4 address used by the PW as the destination IP address of the peer.

Type	Length	Value
9.3	4	ip1.ip2.ip3.ip4

### eTEA PW Peer Address IPv6

This object configures the peer destination IP address when IPv6 is used. This object is required only when IPv6 is used.

Type	Length	Value
9.4	16	ip1....ip16

### eTEA PW Destination Port

This object configures the destination UDP port to be used by the currently indexed PW. The UDP port number shall be chosen from the range of dynamically allocated UDP ports numbers (49152 through 65535). For a given IWF, a unique port number should be assigned for each PW.

Type	Length	Value
9.5	2	<Valid port number>

### eTEA PW Peer Destination Port

This object configures the UDP port used by the peer PW for the currently indexed PW. The UDP port number shall be chosen from the range of dynamically allocated UDP ports numbers (49152 through 65535). This value may also be used as the source UDP port for packets sent to the Peer.

Type	Length	Value
9.6	2	<Valid port number>

### eTEA PW Name

This object stores the canonical name assigned to the PW.

Type	Length	Value
9.7	1-31	<string>

### eTEA PW Description

This object initializes textual string containing information about the PW. This is an optional parameter. If there is no description, this object contains a zero length string.

Type	Length	Value
9.8	1-31	<string>

### eTEA PW Admin Status

This object is used to alter the administrative status of this PW. Testing is used to route the packet side of the PW to the externally accessible Ethernet port, as opposed to the RFI port. This is useful for testing PW operations such as the clock recovery without an intervening DOCSIS network. In some implementations, this may be a global parameter. Application loopback is used to route packets on this PW to itself so that the PW can be looped back on itself. This is useful for testing the TDM interface through the internal data path of the TEA.

Type	Length	Value
9.9	2	up(1),down(2),testing(3), app_loopback(4)

### eTEA PW Status Change Notification Enable

This object configures whether notifications are generated on changes in the status of this PW from up to down or down to up. The default value is Disabled.

Type	Length	Value
9.10	1	Disable (0), Enable (1)

### eTEA PW TDM Type

This object configures the type of TDM link being emulated. When a line type with CAS is selected, this enables the CAS application signalling as described in clause 6.3.2.2.

Type	Length	Value
9.11	1	other(1),ds1(2),e1(3), ds3(4),e3(5),octetAlignedT1(6), nXds0(7),nXds0WithCASe1(8), nXds0WithCASds1Esf(9),nXds0WithCASds1Sf(10)

### eTEA PW TDM Configuration Table Index

This object configures the index to the TDM configuration table. The TDM configuration table contains parameters that are typically common across many PWs of an IWF in an eTEA. This is described by TLV 10 below.

Type	Length	Value
9.12	2	index of PW TDM configuration table

### eTEA PW CESoPSN Configuration Table Index

This object configures the index to the CESoPSN configuration table. This is described by TLV 13 below.

Type	Length	Value
9.13	2	index of CESoPSN specific configuration table

### **eTEA PW RTP SSRC**

In the case where RTP headers are being used on this PW, this object configures the value of the SSRC field to use in the RTP header.

<b>Type</b>	<b>Length</b>	<b>Value</b>
9.14	4	<SSRC value>

### **eTEA PW RTP Peer SSRC**

In the case where RTP headers are being used on this PW, this object configures the value of the SSRC field to expect in the RTP header. When this value is non-zero, it may be used to check incoming data frames for validity. Default: 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
9.15	4	<SSRC value>

### **eTEA PW TDM Circuit Map**

This object configures the TDM circuit transported by this PW. This TLV includes two sub-options to declare the port number and the timeslots. For structure agnostic operation, it configures the port transported by this PW. For structure aware emulation, the 64K timeslots transported by this PW are configured. It is possible for a PW to support timeslots from multiple ports.

<b>Type</b>	<b>Length</b>	<b>Value</b>
9.16	N	

### **eTEA PW TDM Port Number**

This object references the physical E1/T1 port number. The default value is 1, such that in implementations supporting a single port this parameter is not required.

<b>Type</b>	<b>Length</b>	<b>Value</b>
9.16.1	1	Port number (1-255)

### **eTEA PW TDM Timeslot Map**

This object selects the timeslots transported by this PW.

<b>Type</b>	<b>Length</b>	<b>Value</b>
9.16.2	4	Selected timeslots. 32-bit bit map for E1, 24 bit for T1

Example: The value 0x00000605 configures time slots 1, 4, 10, 11; a value of 0x00ffffff configures all 24 timeslots of a T1 interface.

### **eTEA PW TDM Configuration Table**

This object configures the common parameters of a Pseudo Wire configured for TDM Emulation. Table entries may be indexed by multiple pseudo wires.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10	n	

### **eTEA PW TDM Configuration Table Index**

This object configures the current entry of the PW TDM configuration table.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.1	2	<TDM config table index>

### **eTEA PW TDM Payload Size**

This object configures the current payload size used for this PW. In structure agnostic transport, the default value for this parameter is 192 bytes for T1 and 256 bytes for E1. For structure aware transport, this value is the number of timeslots times the number of frames in the payload.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.2	2	Payload size

### **eTEA PW RTP header Used**

This optional object configures the use of RTP header.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.3	1	true(1), false(2)

### **eTEA jitter buffer setting**

This object configures the jitter configuration and is an estimate of maximum jitter in a given network. The jitter buffer is used to counter some of the network congestion and managed network changes which may cause delay variations in the packet arrival rate.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.5	2	<Value in ms>

### **eTEA PW Payload Suppression**

This optional object configures the PW to not send out payload bytes when the L bit is set. The L bit indicates that there is a problem at the TDM interface. Default value is disabled (2).

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.6	1	enable(1),disable(2)

### **eTEA PW LoPS Exit Criteria**

The required number of packets with sequential sequence numbers that are available to be played out of the jitter buffer in order to exit the LoPS state. If set to 0, the LoPS state is exited when the excessive packet loss rate falls below the excessive packet threshold value for a complete window time (AvePktLossTimeWindow).

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.7	2	# consecutive frames

### **eTEA PW LoPS Entrance Criteria**

This object configures the criteria for entering the Loss of packet synchronization state. Indicates the number of consecutive packets that are late or missing on playout to be considered the criteria for entering the LoPS state. When the parameter is set to 0, the LoPS state is entered when the excessive packet loss threshold has been reached. Default value is 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.8	2	# consecutive frames

### **eTEA PW Packet Replace Policy**

This parameter determines the information played out on the TDM interface instead of a lost packet assuming the pseudo wire is not in loss of packet synchronization state. Default value is 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.10	1	filler (0) AIS (1), other (2)

### **eTEA PW Packet Loss Window**

This object configures the size of the Window to be used for calculating the packet loss rate.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.11	2	Ms

### **eTEA PW Excessive Loss Threshold**

This object configures the % threshold to be considered excessive. A second in which this is counted is considered an errored second.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.12	1	%

### **eTEA PW Severe Loss Threshold**

This object configures the % threshold to be considered severe. A second in which this is counted is considered a severe errored second.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.15	1	%

### **eTEA PW RTP Timestamp Mode**

This parameter determines the timestamp generation mode. Default value is 1.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.16	1	absolute(1) differential(2), prime(3), other(4)

### **eTEA PW Default Fill Pattern**

This object configures the fill pattern to play out on the TDM interface when a packet is not present to be played out and the replace policy is to use the configured filler byte.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.17	1	Fill (default 0x00)

### **eTEA PW L Flag Payload Policy**

This object determines whether the payload received in packets.

With L bit set should be ignored or used. Default value is ignore(0).

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.18	1	ignore(0), use(1)

### **eTEA PW ToS**

This object configures the IP ToS value used for packets sent on this PW. The default value is 184 (EF class).

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.19	1	ToS value

### **eTEA PW RTP Payload Type**

This object configures the payload type to use in the RTP header of the data frames.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.20	1	PT value

### **eTEA PW RTP Peer Payload Type**

This object configures the payload type that the peer uses in the RTP header of the data frames. When this value is non-zero, it may be used to check incoming data frames for validity. Default: 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.21	1	PT value

### **eTEA PW RTP Timestamp Reference**

This object configures the timestamp reference to use. This is the number clocks in a 125  $\mu$ s period. For example, the following frequencies are represented as follows: 8k(1), 32k(4), 256k(64), 1.544M(193), 2.048(256), 10.24(1280), 19.44(2430).

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.22	2	TS rate

### **eTEA PW RTP Peer Timestamp Reference**

This object configures the timestamp reference use by peer. This is the number clocks in a 125  $\mu$ s period. For example, the following frequencies are represented as follows: 8k(1), 32k(4), 256k(64), 1.544M(193), 2.048(256), 10.24(1280), 19.44(2430).

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.23	2	TS rate

### **eTEA PW SRTP Enable**

This optional object configures the use of SRTP.

<b>Type</b>	<b>Length</b>	<b>Value</b>
10.24	1	true(1), false(2)

### **eTEA PW TDM CESoPSN Configuration Table**

This object configures the common parameters of a pseudo wire configured for CESoPSN. Table entries may be indexed by multiple pseudo wires.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13	N	

### **eTEA PW CESoPSN Configuration Table Index**

This object configures the current entry of the PW CESoPSN configuration table.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.1	2	<CESoPSN config table index>

### **eTEA PW CESoPSN Default Idle Pattern**

This object configures the default Idle Pattern played out on the TDM interface on various defects, according to policy, including receipt of L bit, packet loss and when the pseudo wire is administratively disabled.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.2	1	idle (default 0xFF)

### **eTEA PW CESoPSN L Flag Policy**

This object configures the action to perform on the TDM interface when packets are received with the L flag set. Default value is idle(0).

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.3	1	idle(0), TrunkAIS(1), idle_code(2)

### **eTEA PW CESoPSN R Flag Policy**

This object configures the action to perform, if any, on the TDM interface when packets are received with the R flag set. For structure agnostic applications, this parameter should be set to 0. Default value is (0).

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.4	1	nothing(0), RAI(1), idle_code(2)

### **eTEA PW CESoPSN Remote Defect Policy**

This object configures the action to perform on the TDM interface when packets are received with the L flag as 0 and the M bits as 01 indicating a Remote Defect. Default value is 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.5	1	Nothing(0), RAI(1), Channel Idle(2)

### **eTEA PW CESoPSN LoPS Policy**

This parameter determines the information played out on the TDM interface while in the LoPS state. This determines if the idle pattern or AIS pattern is played out. For structured aware TDM emulation, 'channel idle' indication on the trunk interface can also be specified.

default: idle(1)

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.6	1	Idle(1), TrunkAIS(2), Channel Idle(3)

### **eTEA PW Application Signalling ToS**

This object configures the value to use in the ToS field when configured to send the CAS signalling in a separate packet stream. The default value is 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.7	1	ToS value

### **eTEA PW RTP CAS Signalling Payload Type**

This object configures the payload type to use in the RTP header of the signalling frames.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.8	1	PT value

### **eTEA PW RTP CAS Signalling Peer Payload Type**

This object configures the payload type to use in the RTP header of the signalling frames. When this value is non-zero, it may be used to check incoming data frames for validity. Default: 0.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.9	1	PT value

### **eTEA PW Application Signalling Idle**

This object configures the CAS signalling Idle pattern. This is the pattern sent when a failure is detected on the TDM interface, including multi-frame failures. This may need to be different per PW. Default 0x0F.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.10	1	Idle value

### **eTEA PW Application Signalling Interval**

This object configures the application signalling interval to use when sending the CAS signalling in a separate packet stream. The default value is 5 ms.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.11	1	ms

### **eTEA PW Application Maximum Signalling Interval**

This object configures the application signalling time in which to refresh the peer when no application signalling has changed in this interval. This is only valid when configured to send the CAS signalling in a separate packet stream. The default value is 5 seconds.

<b>Type</b>	<b>Length</b>	<b>Value</b>
13.12	1	s

### **eTEA dsx1 configuration table**

This object configures the parameters of each dsx1 TDM interface. The parameters correspond to parameters in [IETF RFC 3895]. Status of the interface may be read out using SNMP gets.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14	n	

### **eTEA dsx1 Port ID**

This object configures the PORT ID for the physical dsx1 TDM port that is being configured.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.1	1	<Port ID>

### **eTEA dsx1 Line Type**

This object configures the Line Type for the physical TDM port that is being configured.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.2	1	other(1), dsx1ESF(2), dsx1D4(3), dsx1E1(4), dsx1E1CRC(5), dsx1E1MF(6), dsx1E1CRCMF(7), dsx1Unframed(8), dsx1E1Unframed(9)

### eTEA dsx1 Line Coding

This object configures the Line Coding for the physical TDM port that is being configured.

Type	Length	Value
14.3	1	dsx1JBZS(1), dsx1B8ZS(2), dsx1HDB3(3), dsx1ZBTISI(4), dsx1AMI(5), other(6), dsx1B6ZS(7)

### eTEA dsx1 Circuit ID

This object configures the transmission vendor's circuit identifier to facilitating troubleshooting. This identifier should include the physical port ID.

Type	Length	Value
14.4	1-255	string

### eTEA dsx1 Loopback Configuration

This object configures the Loopback configuration for the physical TDM port that is being configured. This is for diagnostic and commissioning purposes.

Type	Length	Value
14.5	1	dsx1NoLoop(1), dsx1PayloadLoop(2), dsx1LineLoop(3), dsx1OtherLoop(4), dsx1InwardLoop(5), dsx1DualLoop(6)

### eTEA dsx1 Signal Mode

This object configures the signalling mode used for DS0s of this physical port. The default value is none(1).

Type	Length	Value
14.6	1	none(1), robbedBit(2), (CAS Signalling for T1) bitOriented(3), (CAS Signalling for E1) messageOriented(4), (CCS Signalling) other(5)

### eTEA dsx1 Transmit Clock Source

This object configures the clock source for this physical port. Specification of the clock source includes the mode that this port works in along with a reference to the source of the clock, if necessary.

Type	Length	Value
14.7	n	

### eTEA dsx1 Transmit Clock Source Mode

This object configures the clock mode for this physical port. This sub-option must always be present. Other sub-options are optional as described below. All PWs attached to this port work in this timing mode.

Type	Length	Value
14.7.1	1	loopTiming(1), localTiming(2), throughTiming(3), adaptive(4)

When working in loopTiming, the clock source is taken from the received TDM clock of this port. When working in localTiming, the clock source is taken from an unspecified internal clock source. For throughTiming, the clock source is taken from the cable modem interface unless otherwise specified with TLVs 14.7.2 and 14.7.3. For adaptive timing, the clock source is taken from the PW attached to this port. If multiple PWs are associated with this port, the clock comes from the PW attached to the lowest numbered DS0 of this port, unless otherwise specified with TLV 14.7.4.

### eTEA dsx1 Transmit Clock Source Interface Type

This object configures the interface type. The default is 0.

Type	Length	Value
14.7.2	1	Cable Modem port(0); Ethernet port(1); e1/t1 port(2)

### eTEA dsx1 Transmit Clock Source Port Number

This object configures the index of the clock source. The default is 1.

Type	Length	Value
14.7.3	1	Port Number

### eTEA dsx1 Transmit Clock Source PW Index

This object configures the PW index of the adaptive timing source.

Type	Length	Value
14.7.4	2	PW Index

### eTEA dsx1 Fdl

This object configures the set of capabilities desired for the facilities data link on this port. This parameter is only relevant for T1 operation. For structure agnostic emulation, all values can be supported. For structure aware emulation, only a value of dsx1FdlNone is supported.

Type	Length	Value
14.8	1	other(1), dsx1AnsiT1403(2), dsx1Att54016(4), dsx1FdlNone(8)

### **eTEA dsx1 Line Length**

The length of the ds1 line in meters. This objects provides information for line build out circuitry. This object is only useful if the interface has configurable line build out circuitry.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.9	2	Line length (meters)

### **eTEA dsx1 Line Status Trap Enable**

Indicates whether a change in dsx1LineStatus should generate an snmp trap.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.10	1	enabled(1), disabled(2)

### **eTEA dsx1 Channelization**

Indicates whether this DS1 is channelized into separate DS0s.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.11	1	disabled(1), enabledDS0(2)

### **eTEA dsx1 Line Mode**

Indicates whether this DS1 is working as a long or short haul. This is only applicable for T1 and may be useful for configuring the dsx1 interface.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.12	1	csu(1), (long haul) dsu(2), (short haul)

### **eTEA dsx1 Line Build Out**

Indicates the required line build out for this interface port. This is only applicable for T1 and may be useful for configuring the dsx1 interface.

<b>Type</b>	<b>Length</b>	<b>Value</b>
14.13	1	notApplicable(1), neg75dB(2), neg15dB(3), neg225dB(4), zerodB(5)

### **eTEA SNMP MIB Object**

This object allows arbitrary SNMP MIB objects to be Set via the TFTP-Registration process.

<b>Type</b>	<b>Length</b>	<b>Value</b>
11	n	variable binding

where the value is an SNMP VarBind as defined in [IETF RFC 1157]. The VarBind is encoded in ASN.1 Basic Encoding Rules, just as it would be if part of an SNMP Set Request.

The cable modem MUST treat this object as if it were a part of an SNMP Set Request with the following caveats:

- It MUST treat the request as fully authorized (it cannot refuse the request for lack of privilege).

- SNMP Write-Control provisions do not apply.
- No SNMP response needs to be generated.

This object MAY be repeated with different VarBinds to "Set" a number of MIB object. All such SETS MUST be treated as simultaneous.

Each VarBind MUST be limited to 255 bytes.

### eTEA SNMP Write-Access Control

This object makes it possible to disable SNMP "Set" access to individual MIB objects. Each instance of this object controls access to all of the writeable MIB objects whose Object ID (OID) prefix matches. This object may be repeated to disable access to any number of MIB objects.

Type	Length	Value
12	n	OID prefix plus control flag

Where n is the size of the ASN.1 Basic Encoding Rules [ITU-T Y.690] encoding of the OID prefix plus one byte for the control flag.

The control flag may take values:

- 0 – allow write access.
- 1 – disallow write access.

Any OID prefix may be used. The Null OID 0.0 may be used to control access to all MIB objects. When multiple instances of this object are present and overlap, the longest (most specific) prefix has precedence.

Example:

- anyTable disallow write-access.
- anyTable.1.3 allow write-access.

### eTEA Vendor Specific Extensions

This object allows vendor specific extensions to the eTEA. This MUST include the Vendor ID. The Vendor ID MUST be the first TLV. The eTEA MUST ignore any DOCSIS extensions that it cannot interpret.

Type	Length	Value
43	n	

### eTEA Message Integrity Check (eTEA MIC)

This parameter contains a MD5 Digest over the bytes of the config file settings, as outlined in clause 6.7.1.9.4. This TLV appears once in the configuration file immediately before the end of data marker.

Type	Length	Value
6	16	A 128-bit (16 octets) MD5 Digest

### eTEA SNMP Access Control Configuration

The eTEA MUST support the following TLVs from Annex C of [ITU-T J.222.2] for SNMPv1v2c access configuration in SNMPv3 coexistence mode of the eTEA management entity:

- TLV-53 SNMPv1v2c Coexistence Configuration.
- TLV-54 SNMPv3 Access View Configuration.

The eTEA uses the key '@eTEA' instead of '@CM' while populating those entries.

## eTEA End of Data

This is a special marker for end of data. It has neither Length nor value fields.

Type	Length	Value
	255	

### 6.7.1.11 eTEA device capabilities

No eTEA device capability is defined but future enhancement is possible. The hexadecimal fields in DHCP Option 60 should all be set at 0.

### 6.7.2 UGS flow provisioning

For each PW, a UGS flow must be provisioned. The classifier for each UGS flow must include the parameters that uniquely associate the UGS flow to the PW. Each UGS flow should be activated when the PW module in eTEA is enabled, and be deactivated when the PW module in eTEA is disabled.

### 6.7.3 TDM emulation adapter (TEA) provisioning

A TEA may be located either in a CMTS or a router or a switch inside the network. It should have similar parameter sets as an eTEA, but its provisioning is out of the scope of this Recommendation. The TEA MUST support the TEA MIB as shown in Annex B and the next clause. The TEA is configured via SNMP sets or optionally via the configuration file.

## 6.8 Management

Pseudo wires provided by the IWF of the TEA are managed by setting up the configuration, monitoring the statistics provided and responding to the alarms.

Setup and teardown of pseudo wires is based on proper configuration of the peer TEAs that terminate the Pseudo Wire. The OSS is responsible for insuring that parameters of the peer TEAs are compatible.

### 6.8.1 Management model

Several MIB modules are used for providing a standard way of monitoring the operation of the device containing a TEA. The TDM management model consists of several MIB modules, according to the conceptual layering and relationship shown in Figure 6-13 below. The TEA is managed using the MIB modules described [IETF RFC 2494], [IETF RFC 2495], [IETF RFC 2670], [ID PW MIB] and the textual conventions defined in [ID PWTC].

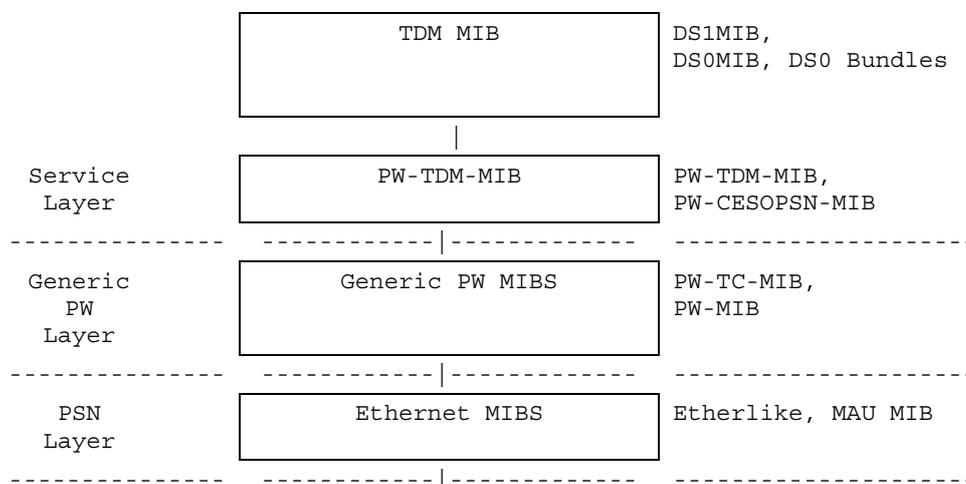


Figure 6-13 – Conceptual layering

NOTE – The various PW MIBs have not yet reached RFC status. As such, a private version of these MIBs has been created as a CableLabs enterprise MIB. These MIBs are described below in general and the ASN.1 is provided as Annex B. The intention is to support the standard MIBs when they reach RFC status. To insure future compatibility with the standard TDM PW MIB when it becomes available, special TLV values have been defined to allow the configuration files to be preserved even when the PW MIBs reach RFC status.

A TDM connection will be pseudo wire connection. It will not be treated as an interface and therefore is not represented in the ifTable.

## 6.8.2 Managed objects requirements

The following clauses detail implementation requirements for the RFCs listed.

### 6.8.2.1 Requirements for [IETF RFC 2494] DS0 and DS0 Bundle MIB

TEAs supporting structure aware transport MUST implement [IETF RFC 2494] for Gets.

No requirement to support the following objects:

```
dsx0RobbedBitSignalling
dsx0CircuitId
dsx0IdleCode
dsx0SeizedCode
dsx0TransmitCodesEnable
```

### 6.8.2.2 Requirements for [IETF RFC 3895] DS1

All TEAs MUST implement [IETF RFC 3895] for Gets.

No requirement to support DS2/E2 interfaces

No requirement to support the following objects:

```
dsx1SendCode
dsx1LoopbackConfig
dsx1SignalMode
dsx1Fdl
dsx1LineLength
dsx1LineCoding
ds1ChanMappingGroup
ds1FarEndGroup
ds1DS2Group
```

When structure aware transport is supported, the TEA MUST support the following parameters:

```
dsx1Channelization
```

The TEA MUST support the following parameter:

dsx1TransmitClockSource using the parameters loop and through. Support for internal is not required.

### 6.8.2.3 Requirements for [IETF RFCs 3635 and 3636] Ethernet MIBs

Support for the [IETF RFC 3635] and [IETF RFC 3636] MIBs is only required when an external Ethernet interface is provided by the TEA. As an eDOCSIS device, the packetized Ethernet interface is a logical interface and its configuration and management is beyond the scope of this Recommendation.

### 6.8.2.4 Requirements for [IETF RFC 2863] If MIB

DOCSIS compliant TEAs MUST implement [IETF RFC 2863].

A TEA MUST support the ifAdminStatus object as R/W to provide administrative control over both TDM and Packet interfaces.

The ifType object has been assigned the following enumerated values for each instance of a TEA interface:

ethernetCsmacd(6),  
 ds1(18),  
 ds0(81),  
 ds0Bundle(82).

#### 6.8.2.4.1 Example use of the ifStackTable

This clause describes by example how to use ifStackTable to represent the relationship of ds0 and ds0Bundles with ds1 interfaces and PWs. Implementers of the stack table for ds0 and ds0Bundle interfaces should use the appropriate RFC for the service being stacked on ds0s and ds0Bundles. Examples given below are for illustration purposes only. Implementation of the ifStackTable and ifStack should be implemented for Gets.

Example: Two PWs, one directly on a ds1, and one PW being carried on four ds0s of a ds1.

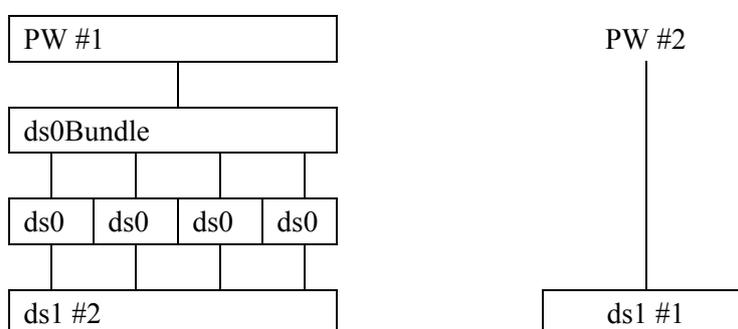


Figure 6-14 – PW examples

The assignment of the index values could, for example, be:

ifIndex	Description	
1	ds1 #1	(type 18)
2	ds0Bundle	(type 82)
3	ds0 #1	(type 81)
4	ds0 #2	(type 81)
5	ds0 #3	(type 81)
6	ds0 #4	(type 81)
7	ds1 #2	(type 18)

The ifStackTable is then used to show the relationships between the various interfaces.

ifStackTable HigherLayer	Entries LowerLayer
0	1
1	0
0	2
2	3
2	4
2	5
2	6
3	7
4	7
5	7
6	7
7	0

The association between PWs and DS0s is configured by the PW circuit map TLV. This allows an implementation to build the ifStackTable and show the relationship based on ifIndexes.

The ifStackTable can be used to define more complicated relationships such as where DS0s from multiple DS1s are grouped together in a DS0 bundle. Application based restrictions are beyond the scope of this Recommendation.

### 6.8.2.5 PW MIB

The private version of the Pseudo Wire MIB is defined in Annex B. This is a snapshot of the current IETF draft along with anticipated changes. This will be rooted in the under DOCSIS projects in the CableLabs private enterprises branch.

### 6.8.2.6 PW TDM MIB

The private version of the Pseudo Wire TDM MIB is defined in Annex B. This is a snapshot of the current IETF draft along with anticipated changes. This will be rooted in the under DOCSIS projects in the CableLabs private enterprises branch.

### 6.8.3 TLV to MIB object mapping

Table 6-10 is the mapping between the TLV options and the MIB Object:

**Table 6-10 – TLV to MIB object mapping**

TLV number	TLV name	MIB object
<b>PW table</b>		
9.1	PW Index	teaPwIndex
9.2	PW Emulation Type	teaPwType
		teaPwPeerAddrType
9.3	PW Peer Addr	teaPwPeerAddr
9.4	PW Peer Addr ipV6	
9.5	PW Dest Port	teaPwOutboundVcLabel
9.6	PW Peer Dest Port	teaPwInboundVcLabel
9.7	PW Name	teaPwName
9.8	PW Description	teaPwDescr
9.9	PW Admin Status	teaPwAdminStatus
9.10	PW Change Notification Enable	teaPwUpDownNotifEnable
9.11	PW TDM Type	teaPwTDMType
9.12	PW TDM config table Index	teaPwGenTDMCfgIndex
9.13	PW TDM CESoPSN table Index	teaPWRelTDMCfgIndex
9.14	PW TDM RTP SSRC	teaPwTDMRtpSSRC
9.15	PW TDM RTP SSRC Peer	teaPwTDMPeerRtpSSRC
9.16	Circuit Map	mapped to DS0 Mib, iftable and ifstack
<b>PW TDM configuration table</b>		
10.1	PW TDM Config Table Index	teaPwTDMCfgIndex
10.2	Payload Size	teaPwTDMCfgPayloadSize
10.3	RTP header used	teaPwTDMCfgRtpHdrUsed
10.5	Jitter Buffer Depth	teaPwTDMCfgJtrBfrDepth

**Table 6-10 – TLV to MIB object mapping**

<b>TLV number</b>	<b>TLV name</b>	<b>MIB object</b>
<b>PW table</b>		
10.6	Payload Suppression	teaPwTDMCfgPayloadSuppression
10.7	LoPS Exit Criteria	teaPwTDMCfgConsecPktsInSynch
10.8	LoPS Entrance Criteria	teaPwTDMCfgConsecMissPktsOutSynch
10.10	Packet Replace Policy	teaPwTDMCfgPktReplacePolicy
10.11	Packet Loss Window	teaPwTDMCfgAvePktLossTimeWindow
10.12	Excessive Loss Threshold	teaPwTDMCfgExcessivePktLossThreshold
10.15	Severe Loss Threshold	teaPwTDMCfgMissingPktsToSes
10.16	RTP Timestamp Mode	teaPwTDMCfgTimestampMode
10.17	Default Fill Pattern	teaPwTDMCfgFillerPattern
10.18	L bit Payload Policy	teaPwTDMCfgLflagPayloadPolicy
10.19	PW IP ToS	teaPwIPTos
10.20	RTP Payload Type	teaPwTDMCfgRtpPT
10.21	RTP Peer Payload Type	teaPwTDMCfgRtpPeerPT
10.22	RTP TS Reference	teaPwTDMCfgRtpTSRef
10.23	RTP Peer TS Reference	teaPwTDMCfgRtpPeerTSRef
10.24	SRTP enable	teaPwTDMCfgSRTPenable
<b>CESoPSN configuration table</b>		
13.1	PW CESoPSN Config Table Index	teaPwTDMCfgFramedIndex
13.2	Default Idle Pattern	teaPwTDMCfgFramedIdlePattern
13.3	LFlag Policy	teaPwTDMCfgFramedLflagPolicy
13.4	RFlag Policy	teaPwTDMCfgFramedRflagPolicy
13.5	Remote Defect Policy	teaPwTDMCfgFramedRDPolicy
13.6	LoPS Policy	teaPwTDMCfgFramedLopsPolicy
13.7	App Sig ToS	teaPwTDMCfgFramedSigIPTos
13.8	RTP CAS PT	teaPwTDMCfgFramedSigPT
13.9	RTP CAS Peer PT	teaPwTDMCfgFramedSigPeerPT
13.10	App Sig Idle	teaPwTDMCfgFramedSigIdle
13.11	App Sig Interval	teaPwTDMCfgFramedSigInterval
13.12	App Sig Max Interval	teaPwTDMCfgFramedSigMaxInterval
<b>dsx1 configuration table</b>		
14.1	dsx1 port ID	physical port number of the ds1 interface
14.2	dsx1 Line type	dsx1LineType
14.3	dsx1 Line Coding	dsx1LineCoding
14.4	dsx1 Circuit ID	dsx1CircuitIdentifier

**Table 6-10 – TLV to MIB object mapping**

<b>TLV number</b>	<b>TLV name</b>	<b>MIB object</b>
<b>PW table</b>		
14.5	dsx1 Loopback Config	dsx1LoopbackConfig
14.6	dsx1 Signal Mode	dsx1SignalMode
14.7	dsx1 Transmit Clock Source	dsx1TransmitClockSource
14.8	dsx1 Fdl	dsx1Fdl
14.9	dsx1 Line Length	dsx1LineLength
14.10	dsx1 Line Status Trap Enable	dsx1LineStatusChangeTrapEnable
14.11	dsx1 Channelization	dsx1Channelization
14.12	dsx1 Line Mode	dsx1LineMode
14.13	dsx1 Line Build Out	dsx1LineBuildOut

## Annex A

### Event, SYSLOG, and SNMP trap extensions

(This annex forms an integral part of this Recommendation)

The TEA MUST support the DOCSIS Event extensions defined in this annex.

This annex follows the format of Annex D (Format and Content for Event, SYSLOG, and SNMP Trap) of [DOCSIS OSSIV2.0]. The TEA MUST conform to the requirements of [DOCSIS OSSIV2.0] clause 7.4, Fault management, pertaining to these events, unless otherwise explicitly indicated in this annex.

#### A.1 TEA events description

"Event" is used in this part to reference Annex D of [DOCSIS OSSIV2.0].

##### A.1.1 TEA event process "Init"

The event process "Init" refers to the initial provisioning process. The Event extensions associated with the "Init" process are divided into two sub-processes, PROVISIONING and CLOCK. The extensions for PROVISIONING use Error Code Set P01, while the extensions for CLOCK use Error Code Set P02.

##### A.1.2 TEA event process "Oper"

The Event extensions herein designated as "Oper" cover events generated during operation. The only defined sub-process is LINK. The Error Code Sets used for these events are P03.

#### A.2 DOCSIS events extensions

**Table A.1 – DOCSIS events extensions**

Process	Sub-process	TEA priority	Event message	Message notes and details	Error code set	Event ID	SNMP notification
<b>INITIALIZATION</b>							
Init	PROVISIONING	Notice	DHCP Broadcast DISCOVER (eTEA-1)				
Init	PROVISIONING	Notice	DHCP OFFER (eTEA-2)				
Init	PROVISIONING	Notice	DHCP broadcast REQUEST (eTEA-3)				

**Table A.1 – DOCSIS events extensions**

Process	Sub-process	TEA priority	Event message	Message notes and details	Error code set	Event ID	SNMP notification
Init	PROVISIONING	Notice	DHCP ACK (eTEA-4)	For SYSLOG only, append: Status = Pass (or Fail).			
Init	PROVISIONING	Notice	TFTP config file request (eTEA-5)				
Init	PROVISIONING	Error	TFTP config file download error (eTEA-6)		P01.0	8000100	
Init	PROVISIONING	Error	TFTP config file error (eTEA-6)		P01.1	8000101	
Init	PROVISIONING	Notice	Provisioning complete (eTEA-7)	For SYSLOG only, append: Status = Pass (or Fail).	P01.3	8000103	linkUp
Init	CLOCK	Notice	Override of clock M/N value by S-CDMA	For SYSLOG only, append: M/N = <P1>/<P2>. P1= M, P2 = N	P02.0	8000200	
Init	CLOCK	Warning	Cannot support clock M/N value	For SYSLOG only, append: M/N = <P1>/<P2>. P1= M, P2 = N	P02.1	8000201	
<b>OPERATION</b>							
Oper	LINK	Error	TDM Data link Status Change.		P03.0	8000300	TDMdata LinkDown
Oper	LINK	Notice	TDM Data link Up.		P03.1	8000301	TDMdata LinkUp
Oper	LINK	Error	TDM Signal link Status Change.		P03.2	8000302	TDMsignal LinkDown
Oper	LINK	Notice	TDM Signal link Up.		P03.3	8000303	TDMsignal LinkUp
Oper	IWF	Error	PW has gone down		P03.4	8000304	PWdown

**Table A.1 – DOCSIS events extensions**

<b>Process</b>	<b>Sub-process</b>	<b>TEA priority</b>	<b>Event message</b>	<b>Message notes and details</b>	<b>Error code set</b>	<b>Event ID</b>	<b>SNMP notification</b>
Oper	IWF	Notice	PW now operational		P03.5	8000305	PWup
Oper	IWF	Notice	PW was deleted		P03.6	8000306	PWdeleted
Oper	IWF	Notice	PW Excessive Loss		P03.7	8000307	PWexcessive Loss
Oper	IWF	Error	PW Severe Loss		P03.8	8000308	PWsevere Loss

## Annex B

### TEA MIB definition

(This annex forms an integral part of this Recommendation)

This annex includes the actual ASN.1 notation for the MIBs.

#### B.1 CL-PW-TC-MIB

```
CL-PW-TC-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, Unsigned32, Integer32
        FROM SNMPv2-SMI          -- [RFC 2578]
    clabProjDocsis
        FROM CLAB-DEF-MIB

TEXTUAL-CONVENTION
    FROM SNMPv2-TC;          -- [RFC 2579]

-- pwStdMIB OBJECT IDENTIFIER ::= { xxx xx }

teaPwTcMIB MODULE-IDENTITY
--    LAST-UPDATED "200507121200Z"
--                                     -- 12 July 2005 12:00:00 GMT
--    ORGANIZATION "Pseudo Wire Edge to Edge Emulation (PWE3) Working
--                  Group"
--    CONTACT-INFO
--    " Thomas D. Nadeau
--      Email: tnadeau@cisco.com
--
--      David Zelig
--      E-mail: davidz@corrigent.com
--
--    The PWE3 Working Group (email distribution pwe3@ietf.org,
--    http://www.ietf.org/html.charters/pwe3-charter.html)
--    "
--
-- DESCRIPTION
--    "Copyright (C) The Internet Society (2005). The
--    initial version of this MIB module was published
--    in RFC 3811. For full legal notices see the RFC
--    itself or see:
--    http://www.ietf.org/copyrights/ianamib.html
--
--    This MIB module defines TEXTUAL-CONVENTIONS
--    for concepts used in Pseudo Wire Edge-to-Edge
--    networks.
--    "
-- Revision history.

-- REVISION "200507121200Z"
--                                     -- 12 July 2005 12:00:00 GMT
-- DESCRIPTION "Initial version published as part of RFC XXXX."

    LAST-UPDATED "200611081400Z " -- Nov 8, 2006
    ORGANIZATION "Cable Television Laboratories, Inc."
    CONTACT-INFO
        "
        Postal: Cable Television Laboratories, Inc.
        858 Coal Creek Circle
        Louisville, Colorado 80027-9750
        U.S.A.
        Phone: +1 303-661-9100
        Fax: +1 303-661-9199
        E-mail: mibs@cablelabs.com"
```

DESCRIPTION  
"This MIB module defines the textual conventions of the Pseudo Wire MIBs. This is to be rooted under the DOCSIS project in the CableLabs private enterprise branch until IANA assignment."

REVISION  
"200611081400Z" -- Nov 8, 2006

DESCRIPTION  
"Changed MIB Module root from clabProjDocsis.14 to clabProjDocsis.17."

REVISION  
"200511111700Z" -- Nov 11, 2005

DESCRIPTION  
"This revision, published as CL-PW-TC-MIB."

::= { clabProjDocsis 14 }

TeaPwGroupID ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"An administrative identification mechanism for grouping a set of service-specific pseudo-wire services. May only have local significance."

SYNTAX Unsigned32

TeaPwIDType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Pseudo-Wire Identifier. Used to identify the PW (together with some other fields) in the signaling session. Zero if the PW is set up manually."

SYNTAX Unsigned32

TeaPwIndexType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Pseudo Wire Index. Locally unique index for indexing several MIB tables associated with a particular PW."

SYNTAX Unsigned32

TeaPwVlanCfg ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"VLAN configuration for Ethernet PW.  
Values between 0 to 4095 indicate the actual VLAN field value.  
A value of 4096 indicates that the object refers to untagged frames, i.e., frames without 802.1Q field.  
A value of 4097 indicates that the object is not relevant."

SYNTAX Integer32 (0..4097)

TeaPwOperStatusTC ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Indicates the operational status of the PW.

- up(1): Ready to pass packets.
- down(2): If PW signaling has not yet finished, or indications available at the service level indicate that the VC is not passing packets.
- testing(3): If AdminStatus at the PW level is set to test.

- dormant(4): The PW is not available because of the required resources are occupied PW with higher priority PWs.
- notPresent(5): Some component is missing to accomplish the set up of the PW.
- lowerLayerDown(6): The underlying PSN or outer tunnel is not in OperStatus 'up' state.

```

"
SYNTAX    INTEGER {
    up(1),
    down(2),
    testing(3),
    unknown(4),
    dormant(5),
    notPresent(6),
    lowerLayerDown(7)
}

```

TeaPwPsnTypeTC ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Identifies the PSN type which the PW will use over the network."

```

SYNTAX    INTEGER {
    mpls      (1),
    l2tp      (2),
    ip        (3),
    mplsOverIp(4),
    gre       (5),
    other     (6)
}

```

TeaPwTypeTC ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Indicates the PW type (i.e., the carried service).

"

```

SYNTAX    INTEGER {
    other(0),
    frameRelayDlci(1),
    atmAal5SduVcc(2),
    atmTransparent(3),
    ethernetTagged(4),
    ethernet(5),
    hdlc(6),
    ppp(7),
    cem(8), -- old format
    atmCellNto1Vcc(9),
    atmCellNto1Vpc(10),
    ipLayer2Transport(11),
    atmCell1to1Vcc(12),
    atmCell1to1Vpc(13),
    atmAal5PduVcc(14),
    frameRelayPortMode(15),
    cep(16),
    e1Satop(17),
    t1Satop(18),
    e3Satop(19),
    t3Satop(20),
    basicCesPsn(21),
    basicTdmIp(22),
    tdmCasCesPsn(23),
    tdmCasTdmIp(24),
    frDlci(25)
}

```

TeaPwAttachmentIdentifierType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"An octet string used in the generalized FEC element for identifying attachment forwarder and groups. The NULL identifier is of zero length.

"

SYNTAX OCTET STRING (SIZE (0..255))

TeaPwCwStatusTC ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Indicates the status of the control word negotiation based on the local configuration and the indications received from the peer node.

waitingForNextMsg(1) indicates that the node is waiting for another label mapping from the remote.

sentWrongBitErrorCode(2) indicates that the local node has notified the peer about mismatch in the C bit.

rxWithdrawWithWrongBitErrorCode(3) indicates that a withdraw message has been received with the wrong C-bit error code.

illegalReceivedBit(4) indicates a C bit configuration with the remote which is not compatible with the PW type.

cwPresent(5) indicates that the CW is present for this PW: if signaling is used - C bit is set and agreed between the peers, and for manually configured PW the local configuration requires the use of the CW.

cwNotPresent(6) indicates that the CW is not present for this PW: if signaling is used - C bit is reset and agreed between the peers, and for manually configured PW the local configuration requires that CW would not be used.

notYetKnown(7) indicates that a label mapping has not yet been received from the peer.

"

SYNTAX INTEGER {  
    waitingForNextMsg (1),  
    sentWrongBitErrorCode (2),  
    rxWithdrawWithWrongBitErrorCode (3),  
    illegalReceivedBit (4),  
    cwPresent (5),  
    cwNotPresent (6),  
    notYetKnown(7)  
}

TeaPwCapabilities ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Indicates the optional capabilities of the control protocol. A value of zero indicates the basic LDP PW signaling. Values may be added in the future based on new capabilities introduced in IETF documents.

"

SYNTAX BITS {  
    pwStatusIndication (0)  
}

TeaPwStatus ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The status of the PW and the interfaces affecting this PW. If none of the bits are set, it indicates no faults are reported.

"

SYNTAX BITS {  
    pwNotForwarding (0),  
    customerFacingPwRxFault (1),  
    customerFacingPwTxFault (2),

```

    psnFacingPwRxFault (3),
    psnFacingPwTxFault (4)
}

TeaPwFragSize ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "If set to value other than zero, it indicates desired
        fragmentation to the value set. If set to zero,
        fragmentation is not desired for PSN bound packets.
        "
    SYNTAX      Unsigned32

TeaPwFragStatus ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "The status of the fragmentation process based on local
        configuration and the remote capability.

        noFrag(0) bit indicates that local configuration is for no
        fragmentation.

        cfgFragGreaterThanPsnMtu(1) bit indicates the local desire
        to fragment, but the fragmentation size desired is greater
        than the MTU available at the PSN between peers.
        Fragmentation is not done in this case.

        cfgFragButRemoteIncapable(2) bit indicates that the local
        configuration indicates the desire for fragmentation but
        the remote is not capable of fragmentation.

        cfgFragFcsLengthMismatch(3) bit indicates that there is a
        mismatch between the FCS size between the local
        configuration and the remote configuration.

        fragEnabled(4) bit indicates that both the local was
        configured for fragmentation and the remote has the
        capability to accept fragmented packets, and the FCS size is
        equal in both peers.
        "
    SYNTAX      BITS {
        noFrag (0),
        cfgFragGreaterThanPsnMtu (1),
        cfgFragButRemoteIncapable (2),
        remoteFragCapable (3),
        fragEnabled (4)
    }

END

```

## B.2 CL-PW-MIB

```

CL-PW-MIB DEFINITIONS ::= BEGIN

IMPORTS
    NOTIFICATION-TYPE, MODULE-IDENTITY, OBJECT-TYPE,
    Integer32, Unsigned32, Counter32, Counter64, TimeTicks
        FROM SNMPv2-SMI
        -- [RFC 2578]

    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF
        -- [RFC 2580]

    TruthValue, RowStatus, StorageType,
    TimeStamp
        FROM SNMPv2-TC
        -- [RFC 2579]

    SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB
        -- [RFC 3411]

```

```

InterfaceIndexOrZero
  FROM IF-MIB -- [RFC 2863]

InetAddressType, InetAddress
  FROM INET-ADDRESS-MIB -- [RFC 4001]

PerfCurrentCount, PerfIntervalCount
  FROM PerfHist-TC-MIB -- [RFC 3593]

HCPperfCurrentCount, HCPperfIntervalCount, HCPperfTimeElapsed,
HCPperfValidIntervals
  FROM HC-PerfHist-TC-MIB -- [RFC 3705]

clabProjDocsis
  FROM CLAB-DEF-MIB

TeaPwTypeTC, TeaPwPsnTypeTC,
TeaPwIndexType, TeaPwGroupID, TeaPwIDType, TeaPwOperStatusTC,
TeaPwAttachmentIdentifierType, TeaPwCwStatusTC, TeaPwCapabilities,
TeaPwStatus, TeaPwFragSize, TeaPwFragStatus
  FROM CL-PW-TC-MIB;

```

teaPwMIB MODULE-IDENTITY

```

LAST-UPDATED "200602070000Z" -- February 7, 2006
ORGANIZATION "Cable Television Laboratories, Inc"
CONTACT-INFO

```

```

"
  Postal: Cable Television Laboratories, Inc.
  858 Coal Creek Circle
  Louisville, Colorado 80027-9750
  U.S.A.
  Phone: +1 303-661-9100
  Fax: +1 303-661-9199
  E-mail: mibs@cablelabs.com"

```

```

DESCRIPTION
  "This version was taken from a snapshot of
  draft-ietf-pwe3-pw--mib-07.txt and modified
  by David Brief of Resolute Networks to be rooted
  in the under DOCSIS projects in the CableLabs
  private enterprises branch. The intention is to
  support the standard MIBs when they reach RFC status."

```

```

REVISION
  "200602070000Z" -- February 7, 2006

```

```

DESCRIPTION
  "This revision, published as CL-PW-MIB."

```

```

 ::= { clabProjDocsis 15 }

```

-- PwStdMIB MODULE-IDENTITY

```

-- LAST-UPDATED "200601110000Z" -- January 11, 2006
-- ORGANIZATION "Pseudo Wire Edge to Edge Emulation (PWE3) Working
-- Group"

```

```

-- CONTACT-INFO

```

```

-- "
-- David Zelig
-- E-mail: davidz@corrigent.com
--

```

```

-- Thomas D. Nadeau
-- Email: tnadeau@cisco.com
--

```

```

-- The PWE3 Working Group (email distribution pwe3@ietf.org,
-- http://www.ietf.org/html.charters/pwe3-charter.html)
--

```

```

-- REVISION

```

```

-- "200601041200Z"
-- -- 4 January 2004 12:00:00 GMT

```

```

-- DESCRIPTION

```

```

--      "Copyright (C) The Internet Society (2006). The initial
--      version of this MIB module was published in RFC XXXX.
-- RFC Editor: Please replace XXXX with RFC number & remove this
-- note.

--      For full legal notices see the RFC itself or see:
--      http://www.ietf.org/copyrights/ianamib.html

--      This MIB module contains managed object definitions for
--      Pseudo Wire operation as in [RFC 3985][RFC 3931][PWCNTRL].
--      architecture', Martini, L., et al, 'Pseudowire Setup and
--      Maintenance using LDP', and Townsley, M., et al, 'Layer Two
--      Tunneling Protocol (Version 3)'.

--      This MIB module enables the use of any underlying packet
--      switched network (PSN). MIB modules that will support
--      PW operations over specific PSN types are defined in
--      separate memos.

--      The indexes for this MIB module are also used to index the
--      PSN-specific tables and the PW-specific tables. The PW Type
--      dictates which PW-specific MIB module to use.
--      "

```

```

-- Top-level components of this MIB.

```

```

-- Notifications

```

```

teaPwNotifications OBJECT IDENTIFIER
                    ::= { teaPwMIB 0 }
--
-- Tables, Scalars
teaPwObjects       OBJECT IDENTIFIER
                    ::= { teaPwMIB 1 }
--
-- Conformance
teaPwConformance  OBJECT IDENTIFIER
                    ::= { teaPwMIB 2 }
--

```

```

-- PW Virtual Connection Table

```

```

teaPwIndexNext OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object contains an appropriate value to be used
        for teaPwIndex when creating entries in the
        teaPwTable. The value 0 indicates that no
        unassigned entries are available. To obtain the
        value of teaPwIndex for a new entry in the
        teaPwTable, the manager issues a management
        protocol retrieval operation to obtain the current
        value of teaPwIndex. After each retrieval
        operation, the agent should modify the value to
        reflect the next unassigned index. After a manager
        retrieves a value, the agent will determine through
        its local policy when this index value will be made
        available for reuse."
    ::= { teaPwObjects 1 }

```

```

teaPwTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TeaPwEntry
    MAX-ACCESS  not-accessible
    STATUS      current

```

DESCRIPTION

"This table specifies information for configuring and status monitoring which are common to all service types and PSN types."

::= { teaPwObjects 2 }

teaPwEntry OBJECT-TYPE

SYNTAX TeaPwEntry  
 MAX-ACCESS not-accessible  
 STATUS current

DESCRIPTION

"A row in this table represents a pseudo wire (PW) virtual connection across a packet network. It is indexed by teaPwIndex, which uniquely identifies a singular connection.

A row is created by the operator or by the agent if the PW is created by non-SNMP application or due to autodiscovery process.

None of the read-create objects values can be changed when teaPwRowStatus is in the active(1) state. Changes are allowed when the teaPwRowStatus is in notInService(2) or notReady(3) states only.

If the operator needs to change one of the values for an active row (i.e. in order to fix a mismatch in configuration between the local node and the peer), the teaPwRowStatus should be first changed to notInService(2), the objects may be changed now, and later to active(1) in order to re-initiate the signaling process with the new values in effect.

"

INDEX { teaPwIndex }  
 ::= { teaPwTable 1 }

TeaPwEntry ::= SEQUENCE {

teaPwIndex	TeaPwIndexType,
teaPwType	TeaPwTypeTC,
teaPwOwner	INTEGER,
teaPwPsnType	TeaPwPsnTypeTC,
teaPwSetUpPriority	Integer32,
teaPwHoldingPriority	Integer32,
teaPwPeerAddrType	InetAddressType,
teaPwPeerAddr	InetAddress,
teaPwAttachedPwIndex	TeaPwIndexType,
teaPwIfIndex	InterfaceIndexOrZero,
teaPwID	TeaPwIDType,
teaPwLocalGroupID	TeaPwGroupID,
teaPwGroupAttachmentID	TeaPwAttachmentIdentifierType,
teaPwLocalAttachmentID	TeaPwAttachmentIdentifierType,
teaPwPeerAttachmentID	TeaPwAttachmentIdentifierType,
teaPwCwPreference	TruthValue,
teaPwLocalIfMtu	Unsigned32,
teaPwLocalIfString	TruthValue,
teaPwLocalCapabAdvert	TeaPwCapabilities,
teaPwRemoteGroupID	TeaPwGroupID,
teaPwCwStatus	TeaPwCwStatusTC,
teaPwRemoteIfMtu	Unsigned32,
teaPwRemoteIfString	SnmpAdminString,
teaPwRemoteCapabilities	TeaPwCapabilities,
teaPwFragmentCfgSize	TeaPwFragSize,
teaPwRmtFragCapability	TeaPwFragStatus,
teaPwFcsRetentioncfg	INTEGER,
teaPwFcsRetentionStatus	BITS,
teaPwOutboundLabel	Unsigned32,
teaPwInboundLabel	Unsigned32,

```

teaPwName          SnmpAdminString,
teaPwDescr         SnmpAdminString,
teaPwCreateTime    TimeStamp,
teaPwUpTime        TimeTicks,
teaPwLastChange    TimeTicks,
teaPwAdminStatus   INTEGER,
teaPwOperStatus    TeaPwOperStatusTC,
teaPwLocalStatus   TeaPwStatus,
teaPwRemoteStatusCapable INTEGER,
teaPwRemoteStatus  TeaPwStatus,
teaPwTimeElapsed   HCPerfTimeElapsed,
teaPwValidIntervals HCPerfValidIntervals,
teaPwRowStatus     RowStatus,
teaPwStorageType   StorageType
}

teaPwIndex OBJECT-TYPE
SYNTAX      TeaPwIndexType
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "Index for the conceptual row identifying a PW within
     this PW Emulation table."
 ::= { teaPwEntry 1 }

teaPwType OBJECT-TYPE
SYNTAX      TeaPwTypeTC
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This value indicates the service to be carried over
     this PW.
    "
 ::= { teaPwEntry 2 }

teaPwOwner OBJECT-TYPE
SYNTAX      INTEGER {
    manual          (1),
    pwIdFecSignaling (2), -- PW signaling with PW ID FEC
    genFecSignaling (3), -- Generalized attachment FEC
    l2tpControlProtocol (4),
    other           (5)
}
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "Set by the operator to indicate the protocol responsible
     for establishing this PW.
     'manual' is used in all cases where no maintenance
     protocol (PW signaling) is used to set up the PW, i.e.,
     require configuration of entries in the PW tables
     including PW labels, etc.
     'pwIdFecSignaling' is used in case of signaling with the
     PwId FEC element with LDP signaling.
     'genFecSignaling' is used of the generalized FEC with LDP
     is used for signaling purposes.
     'l2tpControlProtocol' indicates the use of L2TP
     control protocol.
     'other' is used for other types of signaling."
 ::= { teaPwEntry 3 }

teaPwPsnType OBJECT-TYPE
SYNTAX      TeaPwPsnTypeTC
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "Set by the operator to indicate the PSN type. Based on
     this object, the relevant PSN table entries are created
     in the PSN specific MIB modules.
    "
 ::= { teaPwEntry 4 }

```

```

teaPwSetUpPriority OBJECT-TYPE
    SYNTAX      Integer32 (0..7)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This object defines the relative set-up priority of the PW
        in a lowest-to-highest fashion, where 0 is the highest
        priority. PWs with the same priority are treated with
        equal priority. Dropped PW will be set 'dormant' (as
        indicated in teaPwOperStatus).
        This value is significant if there are competing resources
        between PWs and the implementation supports this feature.
        Equal priority handling with competing resources is
        implementation specific. "
    DEFVAL      { 0 }
    ::= { teaPwEntry 5 }

teaPwHoldingPriority OBJECT-TYPE
    SYNTAX      Integer32 (0..7)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This object defines the relative holding priority of the
        PW in a lowest-to-highest fashion, where 0 is the highest
        priority. PWs with the same priority are treated with
        equal priority. Dropped PW will be set 'dormant' (as
        indicated in teaPwOperStatus).
        This value is significant if there are competing resources
        between PWs and the implementation supports this feature.
        Equal priority handling with competing resources is
        implementation specific. "
    DEFVAL      { 0 }
    ::= { teaPwEntry 6 }

teaPwPeerAddrType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Denotes the address type of the peer node. It should be
        set to 'unknown' if PE/PW maintenance protocol is not used
        and the address is unknown."
    DEFVAL      { ipv4 }
    ::= { teaPwEntry 8 }

teaPwPeerAddr OBJECT-TYPE
    SYNTAX      InetAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This object contains the value of the peer node address
        of the PW/PE maintenance protocol entity. This object
        SHOULD contain a value of all zeroes if not applicable
        (teaPwPeerAddrType is 'unknown')."
    ::= { teaPwEntry 9 }

teaPwAttachedPwIndex OBJECT-TYPE
    SYNTAX      TeaPwIndexType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "If the PW is attached to another PW instead of a local
        native service, this item indicates the teaPwIndex of the
        attached PW, and the PW specific entry in the service MIB
        module MUST not be created. Otherwise, this object MUST
        be set to zero."
    DEFVAL      { 0 }
    ::= { teaPwEntry 10 }

```

```

teaPwIfIndex OBJECT-TYPE
    SYNTAX      InterfaceIndexOrZero
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "If the PW is represented in the ifTable as an ifIndex, this
         item holds the ifIndex of the PW in the ifTable, otherwise
         it MUST be set to zero."
    DEFVAL { 0 }
    ::= { teaPwEntry 11 }

teaPwID OBJECT-TYPE
    SYNTAX      TeaPwIDType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Used in the outgoing PW ID field within the 'Virtual
         Circuit FEC Element'.
         Applicable if teaPwOwner equal 'pwIdFecSignaling' or
         'l2tpControlProtocol', should be set to zero otherwise."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol.'"
    ::= { teaPwEntry 12 }

teaPwLocalGroupID OBJECT-TYPE
    SYNTAX      TeaPwGroupID
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Used in the Group ID field sent to the peer PWEs
         within the maintenance protocol used for PW setup.
         Applicable if teaPwOwner equal 'pwIdFecSignaling' or
         'l2tpControlProtocol', should be set to zero otherwise."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol.'"
    ::= { teaPwEntry 13 }

teaPwGroupAttachmentID OBJECT-TYPE
    SYNTAX      TeaPwAttachmentIdentifierType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This object is an octet string representing the attachment
         group identifier (AGI) this PW belongs to, which typically
         identifies the VPN ID.
         Applicable if teaPwOwner equal 'genFecSignaling'."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol.'"
    ::= { teaPwEntry 14 }

teaPwLocalAttachmentID OBJECT-TYPE
    SYNTAX      TeaPwAttachmentIdentifierType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This object is an octet string representing the local
         forwarder attachment individual identifier (AII) to be
         used by this PW. It is used as the SAII for outgoing
         signaling messages and the TAIID in the incoming messages
         from the peer.

         Applicable if teaPwOwner equal 'genFecSignaling'."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol.'"
    ::= { teaPwEntry 15 }

```

```

teaPwPeerAttachmentID OBJECT-TYPE
    SYNTAX      TeaPwAttachmentIdentifierType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This object is an octet string representing the peer
        forwarder attachment individual identifier (AII) to be
        used by this PW. It is used as the TAI for outgoing
        signaling messages and the SAI in the incoming messages
        from the peer.
        Applicable if teaPwOwner equal 'genFecSignaling'. "
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
        the Label Distribution Protocol.'"
 ::= { teaPwEntry 16 }

teaPwCwPreference OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Defines if the control word will be sent with each packet
        by the local node. Some PW types mandate the use of
        control word, and in such cases the value configured has
        no effect on the existence of the control word."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
        the Label Distribution Protocol.'"
    DEFVAL { false }
 ::= { teaPwEntry 17 }

teaPwLocalIfMtu OBJECT-TYPE
    SYNTAX      Unsigned32 (0..65535)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "If not equal zero, the optional IfMtu object in the
        signaling protocol will be sent with this value,
        representing the locally supported MTU size over the
        interface (or the virtual interface) associated with the
        PW."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
        the Label Distribution Protocol.'"
    DEFVAL { 0 }
 ::= { teaPwEntry 18 }

teaPwLocalIfString OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A PW MAY be associated to an interface (or a virtual
        interface) in the ifTable of the node as part of the
        service configuration. This object defines if the
        maintenance protocol will send the interface's name as
        appears on the ifTable in the name object as part of the
        maintenance protocol. If set to false, the optional
        element will not be sent."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
        the Label Distribution Protocol.'"
    DEFVAL { false }
 ::= { teaPwEntry 19 }

teaPwLocalCapabAdvert OBJECT-TYPE
    SYNTAX      TeaPwCapabilities
    MAX-ACCESS  read-create
    STATUS      current

```

```

DESCRIPTION
    "Indicates the capabilities to be advertised by this node
    to the peer."
REFERENCE
    "Martini, et al, 'Pseudowire Setup and Maintenance using
    the Label Distribution Protocol.'"
 ::= { teaPwEntry 20 }

teaPwRemoteGroupID OBJECT-TYPE
SYNTAX      TeaPwGroupID
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Obtained from the Group ID field as received via the
    maintenance protocol used for PW setup, zero if not used.
    Value of 0xFFFF shall be used if the object is yet to be
    defined by the PW maintenance protocol."
REFERENCE
    "Martini, et al, 'Pseudowire Setup and Maintenance using
    the Label Distribution Protocol.'"
 ::= { teaPwEntry 21 }

teaPwCwStatus OBJECT-TYPE
SYNTAX      TeaPwCwStatusTC
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "If signaling is used for PW establishment, this object
    indicates the status of the control word negotiation,
    and in both signaling or manual configuration indicates
    if CW is to be present or not for this PW."
REFERENCE
    "Martini, et al, 'Pseudowire Setup and Maintenance using
    the Label Distribution Protocol.'"
 ::= { teaPwEntry 22 }

teaPwRemoteIfMtu OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The remote interface MTU as (optionally) received from the
    remote node via the maintenance protocol. Should be zero
    if this parameter is not available or not used."
REFERENCE
    "Martini, et al, 'Pseudowire Setup and Maintenance using
    the Label Distribution Protocol.'"
 ::= { teaPwEntry 23 }

teaPwRemoteIfString OBJECT-TYPE
SYNTAX      SnmpAdminString (SIZE (0..80))
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Indicates the interface description string as received by
    the maintenance protocol, MUST be NULL string if not
    applicable or not known yet."
REFERENCE
    "Martini, et al, 'Pseudowire Setup and Maintenance using
    the Label Distribution Protocol.'"
 ::= { teaPwEntry 24 }

teaPwRemoteCapabilities OBJECT-TYPE
SYNTAX      TeaPwCapabilities
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Indicates the capabilities as advertised by the peer."
REFERENCE
    "Martini, et al, 'Pseudowire Setup and Maintenance using
    the Label Distribution Protocol.'"

```

```

 ::= { teaPwEntry 25 }

teaPwFragmentCfgSize OBJECT-TYPE
    SYNTAX      TeaPwFragSize
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "If set to value other than zero, indicates that
         fragmentation is desired for this PW."
    REFERENCE
        "Malis A., Townsley M., 'PWE3 Fragmentation and
         Reassembly'."
    DEFVAL { 0 } -- i.e., fragmentation not desired
    ::= { teaPwEntry 26 }

teaPwRmtFragCapability OBJECT-TYPE
    SYNTAX      TeaPwFragStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The status of the fragmentation based on the local
         configuration and the peer capabilities as advertised by
         the peer if control protocol is used."
    REFERENCE
        "Malis A., Townsley M., 'PWE3 Fragmentation and
         Reassembly'."
    ::= { teaPwEntry 27 }

teaPwFcsRetentioncfg OBJECT-TYPE
    SYNTAX      INTEGER {
                    fcsRetentionDisable (1),
                    fcsRetentionEnable  (2)
                }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The local configuration of FCS retention for this PW. FCS
         retention can be configured for PW types HDLC, PPP and
         Ethernet only. If the agent does not support FCS retention,
         the error should be configured in teaPwFcsRetentionStatus."
    REFERENCE
        "Malis A., et al., 'PWE3 Frame Check Sequence Retention'
         - work in progress"
    DEFVAL { fcsRetentionDisable }
    ::= { teaPwEntry 28 }

teaPwFcsRetentionStatus OBJECT-TYPE
    SYNTAX      BITS {
                    remoteIndicationUnknown (0),
                    remoteRequestFcsRetention (1),
                    fcsRetentionEnabled (2),
                    fcsRetentionDisabled (3),
                    localFcsRetentionCfgErr (4),
                    fcsRetentionFcsSizeMismatch (5)
                }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The status of the FCS retention negotiation process based on
         local configuration and the remote advertisement.
         remoteIndicationUnknown - set if a FEC has not been received
         from the remote.
         remoteRequestFcsRetention - indicates that the peer has
         requested for FCS retention. FCS retention will be used if
         the local node is capable and configured to use it for this
         PW.
         fcsRetentionEnabled - FCS retention is enabled (both peers
         were configured for FCS retention for signaled PW, or the
         local node is configured and capable for FCS retention).
         fcsRetentionDisabled - FCS retention is disabled (not
         configured locally or not advertised by the peer).

```

```

    localFcsRetentionCfgErr - Set if the local node has been
    configured for FCS retention but is not capable to support
    it.
    fcsRetentionFcsSizeMismatch - Set if there is a FCS size
    mismatch between the local and the peer node.
    "
REFERENCE
    "Malis A., et al., 'PWE3 Frame Check Sequence Retention'
    - work in progress"
::= { teaPwEntry 29 }

teaPwOutboundLabel OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The PW label used in the outbound direction (i.e., toward
        the PSN). It may be set up manually if owner is 'manual'
        or automatically otherwise. Examples: For MPLS PSN, it
        represents the 20 bits of PW tag, for L2TP it represents
        the 32 bits Session ID.
        If the label is not yet known (signaling in process), the
        object should return a value of 0xFFFF."
::= { teaPwEntry 30 }

teaPwInboundLabel OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The PW label used in the inbound direction (i.e., packets
        received from the PSN). It may be set up manually if owner
        is 'manual' or automatically otherwise.
        Examples: For MPLS PSN, it represents the 20 bits of PW
        tag, for L2TP it represents the 32 bits Session ID.
        If the label is not yet known (signaling in process), the
        object should return a value of 0xFFFF."
::= { teaPwEntry 31 }

teaPwName OBJECT-TYPE
    SYNTAX      SnmpAdminString
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The canonical name assigned to the PW."
::= { teaPwEntry 32 }

teaPwDescr OBJECT-TYPE
    SYNTAX      SnmpAdminString
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "A textual string containing information about the PW.
        If there is no description this object contains a zero
        length string."
::= { teaPwEntry 33 }

teaPwCreateTime OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "System time when this PW was created."
::= { teaPwEntry 34 }

teaPwUpTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current

```

```

DESCRIPTION
    "Specifies the total time this PW operStatus has been
    Up(1)."
```

```

 ::= { teaPwEntry 35 }
```

```

teaPwLastChange OBJECT-TYPE
SYNTAX          TimeTicks
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The value of sysUpTime at the time the PW entered
    its current operational state.  If the current state was
    entered prior to the last re-initialization of the local
    network management subsystem, then this object contains a
    zero value."
```

```

 ::= { teaPwEntry 36 }
```

```

teaPwAdminStatus OBJECT-TYPE
SYNTAX          INTEGER {
                    up(1),      -- ready to pass packets
                    down(2),
                    testing(3) -- in some test mode
                }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The desired operational status of this PW."
```

```

 ::= { teaPwEntry 37 }
```

```

teaPwOperStatus OBJECT-TYPE
SYNTAX          TeaPwOperStatusTC
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "This object indicates the operational status of the PW, it
    does not reflect the status of the CE bound interface.
    It is set to down only if pwNotForwarding,
    psnFacingPwRxFault, or psnFacingPwTxFault indications are
    set in teaPwLocalStatus or teaPwRemoteStatus.
    It indicates 'lowerLayerDown' if the only reason for
    not being in the 'up' state is either outer tunnel
    or physical layer down of the network side is in the down
    state.
    All other states are declared based on the description in
    the textual convention.
    "
```

```

 ::= { teaPwEntry 38 }
```

```

teaPwLocalStatus OBJECT-TYPE
SYNTAX          TeaPwStatus
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Indicates the status of the PW in the local node.
    The various indications in this object should be
    available independent of the ability of the local node to
    advertise them or the remote node to accept these status
    indications through the control protocol.
    "
```

```

 ::= { teaPwEntry 39 }
```

```

teaPwRemoteStatusCapable OBJECT-TYPE
SYNTAX          INTEGER {
                    notApplicable (1),
                    notYetKnown   (2),
                    remoteCapable  (3),
                    remoteNotCapable (4)
                }
MAX-ACCESS      read-only
STATUS          current
```

```

DESCRIPTION
    "Indicates the remote node capability to advertise the
    PW status notification.
    notApplicable should be reported for manually set PW, or
    if the local node is not capable of sending the status
    notification object.
    notYetKnown should be reported if the signaling protocol
    has not yet finished the process of capability
    determination.
    remoteCapable and remoteNotcapable should be reported
    based on the initial signaling exchange that has
    determined the remote node capability.
    "
 ::= { teaPwEntry 40 }

teaPwRemoteStatus OBJECT-TYPE
    SYNTAX      TeaPwStatus
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Indicates the status of the PW as was advertised by the
        remote. If the remote is not capable of advertising the
        status object, or the local node is not able to accept
        the status object through signaling, then the applicable
        bit is 'pwNotForwarding' which is set if the remote has
        sent label release or label withdraw for this PW.
        "
 ::= { teaPwEntry 41 }

teaPwTimeElapsed OBJECT-TYPE
    SYNTAX      HCPperfTimeElapsed
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of seconds, including partial seconds,
        that have elapsed since the beginning of the current
        interval measurement period."
 ::= { teaPwEntry 42 }

teaPwValidIntervals OBJECT-TYPE
    SYNTAX      HCPperfValidIntervals
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of previous 15-minute intervals
        for which data was collected."
 ::= { teaPwEntry 43 }

teaPwRowStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "For creating, modifying, and deleting this row. None of the
        read-create objects in the conceptual rows may be changed
        when this object is in the active(1) state."
 ::= { teaPwEntry 44 }

teaPwStorageType OBJECT-TYPE
    SYNTAX      StorageType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "This variable indicates the storage type for this
        object."
 ::= { teaPwEntry 45 }

-- End of PW Virtual Connection Table

-- PW Performance Table.

```

```

teaPwPerfCurrentTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TeaPwPerfCurrentEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table provides per-PW performance information for
        the current interval."
    ::= { teaPwObjects 3 }

teaPwPerfCurrentEntry OBJECT-TYPE
    SYNTAX      TeaPwPerfCurrentEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in this table is created by the agent for
        every PW."
    INDEX { teaPwIndex }
    ::= { teaPwPerfCurrentTable 1 }

TeaPwPerfCurrentEntry ::= SEQUENCE {
    teaPwPerfCurrentInHCPackets      HCPerfCurrentCount,
    teaPwPerfCurrentInHCBytes       HCPerfCurrentCount,
    teaPwPerfCurrentOutHCPackets    HCPerfCurrentCount,
    teaPwPerfCurrentOutHCBytes      HCPerfCurrentCount,
    teaPwPerfCurrentInPackets       PerfCurrentCount,
    teaPwPerfCurrentInBytes         PerfCurrentCount,
    teaPwPerfCurrentOutPackets      PerfCurrentCount,
    teaPwPerfCurrentOutBytes        PerfCurrentCount
}

teaPwPerfCurrentInHCPackets OBJECT-TYPE
    SYNTAX      HCPerfCurrentCount
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "High capacity counter for number of packets received by
        the PW (from the PSN) in the current 15-minute interval.
        This is the 64-bit version of teaPwPerfCurrentInPackets,
        if teaPwPerfCurrentInHCPackets is supported according to
        the rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 1 }

teaPwPerfCurrentInHCBytes OBJECT-TYPE
    SYNTAX      HCPerfCurrentCount
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "High capacity counter for number of bytes received by the
        PW (from the PSN) in the current 15-minute interval.
        This is the 64-bit version of teaPwPerfCurrentInBytes, if
        teaPwPerfCurrentInHCBytes is supported according to the
        rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 2 }

teaPwPerfCurrentOutHCPackets OBJECT-TYPE
    SYNTAX      HCPerfCurrentCount
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "High capacity counter for number of packets forwarded by
        the PW (to the PSN) in the current 15-minute interval.
        This is the 64-bit version of teaPwPerfCurrentOutPackets,
        if teaPwPerfCurrentOutHCPackets is supported according to
        the rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 3 }

teaPwPerfCurrentOutHCBytes OBJECT-TYPE
    SYNTAX      HCPerfCurrentCount
    MAX-ACCESS  read-only
    STATUS      current

```

```

DESCRIPTION
    "High capacity counter for number of bytes forwarded by
    the PW (to the PSN) in the current 15-minute interval.
    This is the 64-bit version of teaPwPerfCurrentOutBytes,
    if teaPwPerfCurrentOutHCBytes is supported according to
    the rules spelled out in RFC 2863."
::= { teaPwPerfCurrentEntry 4 }

teaPwPerfCurrentInPackets OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter for number of packets received by the PW (from
        the PSN) in the current 15-minute interval.
        This is the 32-bit version of teaPwPerfCurrentInHCPackets,
        if teaPwPerfCurrentInHCPackets is supported according to
        the rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 5 }

teaPwPerfCurrentInBytes OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter for number of bytes received by the
        PW (from the PSN) in the current 15-minute interval.
        It MUST be equal to the least significant 32 bits of
        teaPwPerfCurrentInHCBytes, if teaPwPerfCurrentInHCBytes is
        supported according to the rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 6 }

teaPwPerfCurrentOutPackets OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter for number of packets forwarded by
        the PW (to the PSN) in the current 15-minute interval.
        It MUST be equal to the least significant 32 bits of
        teaPwPerfCurrentOutHCPackets, if
        teaPwPerfCurrentOutHCPackets is supported according to the
        rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 7 }

teaPwPerfCurrentOutBytes OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter for number of bytes forwarded by
        the PW (to the PSN) in the current 15-minute interval.
        It MUST be equal to the least significant 32 bits of
        teaPwPerfCurrentOutHCBytes, if teaPwPerfCurrentOutHCBytes is
        supported according to the rules spelled out in RFC 2863."
    ::= { teaPwPerfCurrentEntry 8 }

-- End of PW Perf current Table

-- PW Performance Interval Table.

teaPwPerfIntervalTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF TeaPwPerfIntervalEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table provides per-PW performance information for
        each interval."
    ::= { teaPwObjects 4 }

```

```

teaPwPerfIntervalEntry OBJECT-TYPE
    SYNTAX      TeaPwPerfIntervalEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in this table is created by the agent for every
        PW."
    INDEX { teaPwIndex, teaPwPerfIntervalNumber }
    ::= { teaPwPerfIntervalTable 1 }

TeaPwPerfIntervalEntry ::= SEQUENCE {
    teaPwPerfIntervalNumber      Integer32,
    teaPwPerfIntervalValidData   TruthValue,
    teaPwPerfIntervalTimeElapsed HCPerfTimeElapsed,
    teaPwPerfIntervalInHCPackets HCPerfIntervalCount,
    teaPwPerfIntervalInHCBytes   HCPerfIntervalCount,
    teaPwPerfIntervalOutHCPackets HCPerfIntervalCount,
    teaPwPerfIntervalOutHCBytes  HCPerfIntervalCount,
    teaPwPerfIntervalInPackets   PerfIntervalCount,
    teaPwPerfIntervalInBytes     PerfIntervalCount,
    teaPwPerfIntervalOutPackets  PerfIntervalCount,
    teaPwPerfIntervalOutBytes    PerfIntervalCount
}

teaPwPerfIntervalNumber OBJECT-TYPE
    SYNTAX      Integer32 (1..96)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A number N, between 1 and 96, which identifies the
        interval for which the set of statistics is available.
        The interval identified by 1 is the most recently
        completed 15-minute interval, and the interval identified
        by N is the interval immediately preceding the one
        identified by N-1.
        The minimum range of N is 1 through 4. The default range
        is 1 to 32. The maximum range of N is 1 through 96. "
    REFERENCE
        "Tesink, K. 'Definitions of Managed Objects for the
        SONET/SDH Interface Type', RFC 2558"
    ::= { teaPwPerfIntervalEntry 1 }

teaPwPerfIntervalValidData OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This variable indicates if the data for this interval
        is valid."
    ::= { teaPwPerfIntervalEntry 2 }

teaPwPerfIntervalTimeElapsed OBJECT-TYPE
    SYNTAX      HCPerfTimeElapsed
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The duration of this interval in seconds."
    ::= { teaPwPerfIntervalEntry 3 }

teaPwPerfIntervalInHCPackets OBJECT-TYPE
    SYNTAX      HCPerfIntervalCount
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "High capacity counter for number of packets received by
        the PW (from the PSN) during the interval. This is the 64-bit
        version of teaPwPerfIntervalInPackets, if
        teaPwPerfIntervalInHCPackets is supported according to the
        rules spelled out in RFC 2863."
    ::= { teaPwPerfIntervalEntry 4 }

```

```

teaPwPerfIntervalInHCBytes OBJECT-TYPE
    SYNTAX          HCPerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "High capacity counter for number of bytes received by the
        PW (from the PSN) during the interval.
        This is the 64-bit version of teaPwPerfIntervalInBytes, if
        teaPwPerfIntervalInHCBytes is supported according to the
        rules spelled out in RFC 2863."
    ::= { teaPwPerfIntervalEntry 5 }

teaPwPerfIntervalOutHCPackets OBJECT-TYPE
    SYNTAX          HCPerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "High capacity counter for number of packets forwarded by
        the PW (to the PSN) during the interval.
        This is the 64-bit version of teaPwPerfIntervalOutPackets,
        if teaPwPerfIntervalOutHCPackets is supported according to
        the rules spelled out in RFC 2863."
    ::= { teaPwPerfIntervalEntry 6 }

teaPwPerfIntervalOutHCBytes OBJECT-TYPE
    SYNTAX          HCPerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "High capacity counter for number of bytes forwarded by
        the PW (to the PSN) during the interval.
        This is the 64-bit version of teaPwPerfIntervalOutBytes,
        if teaPwPerfIntervalOutHCBytes is supported according to
        the rules spelled out in RFC 2863."
    ::= { teaPwPerfIntervalEntry 7 }

teaPwPerfIntervalInPackets OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This value represents the number of packets received
        by this PW during the interval.
        It MUST be equal to the least significant 32 bits of
        teaPwPerfIntervalInHCPackets if teaPwPerfIntervalInHCPackets
        is supported according to the rules spelled out in
        RFC 2863."
    ::= { teaPwPerfIntervalEntry 8 }

teaPwPerfIntervalInBytes OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This value represents the number of bytes received
        by this PW during the interval.
        It MUST be equal to the least significant 32 bits of
        teaPwPerfIntervalInHCBytes if teaPwPerfIntervalInHCBytes is supported
        according to the rules spelled out in RFC 2863."
    ::= { teaPwPerfIntervalEntry 9 }

teaPwPerfIntervalOutPackets OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This value represents the number of packets sent by this
        PW during the interval.
        It MUST be equal to the least significant 32 bits of
        teaPwPerfIntervalOutHCPackets if
        teaPwPerfIntervalOutHCPackets is supported according to the

```

```

        rules spelled out in RFC 2863."
 ::= { teaPwPerfIntervalEntry 10 }

teaPwPerfIntervalOutBytes OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This value represents the number of bytes sent by this
        PW during the interval.
        It MUST be equal to the least significant 32
        bits of teaPwPerfIntervalOutHCBYtes
        if teaPwPerfIntervalOutHCBYtes is supported according to
        the rules spelled out in RFC 2863."
 ::= { teaPwPerfIntervalEntry 11 }

-- End of PW Performance Interval Table

-- PW Performance 1 Day Interval Table.

teaPwPerf1DayIntervalTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF TeaPwPerf1DayIntervalEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table provides per-PW Performance information for
        the current day measurement and the previous days
        interval."
 ::= { teaPwObjects 5 }

teaPwPerf1DayIntervalEntry OBJECT-TYPE
    SYNTAX          TeaPwPerf1DayIntervalEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created by the agent for every
        PW."
    INDEX           { teaPwIndex, teaPwPerf1DayIntervalNumber }

 ::= { teaPwPerf1DayIntervalTable 1 }

TeaPwPerf1DayIntervalEntry ::= SEQUENCE {
    teaPwPerf1DayIntervalNumber          Unsigned32,
    teaPwPerf1DayIntervalValidData      TruthValue,
    teaPwPerf1DayIntervalMoniSecs       HCPerfTimeElapsed,
    teaPwPerf1DayIntervalInHCPackets    Counter64,
    teaPwPerf1DayIntervalInHCBytes     Counter64,
    teaPwPerf1DayIntervalOutHCPackets   Counter64,
    teaPwPerf1DayIntervalOutHCBytes     Counter64
}

teaPwPerf1DayIntervalNumber OBJECT-TYPE
    SYNTAX          Unsigned32(1..31)
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "History Data Interval number. Interval 1 is the current day
        measurement period, Interval 2 is the most recent previous
        day; interval 30 is 31 days ago. Intervals 3..31 are
        optional."
 ::= { teaPwPerf1DayIntervalEntry 1 }

teaPwPerf1DayIntervalValidData OBJECT-TYPE
    SYNTAX          TruthValue
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This variable indicates if the data for this interval
        is valid."
 ::= { teaPwPerf1DayIntervalEntry 2 }

```

```

teaPwPerf1DayIntervalMoniSecs OBJECT-TYPE
    SYNTAX      HCPperfTimeElapsed
    UNITS        "seconds"
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The amount of time in the 1-day interval over which the
        performance monitoring information is actually counted.
        This value will be the same as the interval duration except
        in a situation where performance monitoring data could not
        be collected for any reason or agent clock adjustments."
    ::= { teaPwPerf1DayIntervalEntry 3 }

teaPwPerf1DayIntervalInHCPackets OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "High capacity counter for the total number of packets
        received by the PW (from the PSN)."
    ::= { teaPwPerf1DayIntervalEntry 4 }

teaPwPerf1DayIntervalInHCBytes OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "High capacity counter for the total number of bytes
        received by the PW (from the PSN)."
    ::= { teaPwPerf1DayIntervalEntry 5 }

teaPwPerf1DayIntervalOutHCPackets OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "High capacity counter for the total number of packets
        forwarded by the PW (to the PSN)."
    ::= { teaPwPerf1DayIntervalEntry 6 }

teaPwPerf1DayIntervalOutHCBytes OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "High capacity counter for the total number of bytes
        forwarded by the PW (to the PSN)."
    ::= { teaPwPerf1DayIntervalEntry 7 }

-- End of PW Perf 1 Day Interval Table

-- Error counter scalar

teaPwPerfTotalErrorPackets OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Counter for number of error at the PW processing level,
        i.e., packets received with unknown PW label."
    ::= { teaPwObjects 6 }

-- Reverse mapping tables

-- The PW ID mapping table
teaPwIndexMappingTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TeaPwIndexMappingEntry
    MAX-ACCESS   not-accessible
    STATUS       current

```

```

DESCRIPTION
    "This table enables the reverse mapping the unique PwId
    parameters <peer IP, PW type and PW ID> and the
    teaPwIndex. The table is not applicable for PW created
    manually or by using the generalized FEC."
 ::= { teaPwObjects 7 }

teaPwIndexMappingEntry OBJECT-TYPE
    SYNTAX      TeaPwIndexMappingEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in this table MUST be created by the agent for
        every PW created by the teaPwTable for which teaPwOwner
        equals teaPwIdFecSignaling."

    INDEX { teaPwIndexMappingPwType, teaPwIndexMappingPwID,
            teaPwIndexMappingPeerAddrType, teaPwIndexMappingPeerAddr
            }
 ::= { teaPwIndexMappingTable 1 }

TeaPwIndexMappingEntry ::= SEQUENCE {
    teaPwIndexMappingPwType      TeaPwTypeTC,
    teaPwIndexMappingPwID       TeaPwIDType,
    teaPwIndexMappingPeerAddrType InetAddressType,
    teaPwIndexMappingPeerAddr   InetAddress,
    teaPwIndexMappingPwIndex    TeaPwIndexType
}

teaPwIndexMappingPwType OBJECT-TYPE
    SYNTAX      TeaPwTypeTC
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The PW type (indicates the service) of this PW."
 ::= { teaPwIndexMappingEntry 1 }

teaPwIndexMappingPwID OBJECT-TYPE
    SYNTAX      TeaPwIDType
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The PW ID of this PW. Zero if the PW is configured
        manually."
 ::= { teaPwIndexMappingEntry 2 }

teaPwIndexMappingPeerAddrType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "IP address type of the peer node."
 ::= { teaPwIndexMappingEntry 3 }

teaPwIndexMappingPeerAddr OBJECT-TYPE
    SYNTAX      InetAddress
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "IP address type of the peer node."
 ::= { teaPwIndexMappingEntry 4 }

teaPwIndexMappingPwIndex OBJECT-TYPE
    SYNTAX      TeaPwIndexType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value that represents the PW in the teaPwTable."
 ::= { teaPwIndexMappingEntry 5 }

-- End of the PW ID mapping table

```

```

-- The peer mapping table

teaPwPeerMappingTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF TeaPwPeerMappingEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table provides reverse mapping of the existing PW
         based on PW type and PW ID ordering. This table is
         typically useful for EMS ordered query of existing PWs."
    ::= { teaPwObjects 8 }

teaPwPeerMappingEntry OBJECT-TYPE
    SYNTAX          TeaPwPeerMappingEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created by the agent for every
         PW configured in teaPwTable.

        Implementers need to be aware that if the value of
        the teaPwPeerMappingPeerAddr (an OID) or
        teaPwPeerMappingPwID (an OID) (or both combined)
        has more than 111 sub-identifiers, then OIDs of column
        instances in this table will have more than 128
        sub-identifiers and cannot be accessed using SNMPv1,
        SNMPv2c, or SNMPv3."
    INDEX { teaPwPeerMappingPeerAddrType, teaPwPeerMappingPeerAddr,
            teaPwPeerMappingPwType, teaPwPeerMappingPwID }

    ::= { teaPwPeerMappingTable 1 }

TeaPwPeerMappingEntry ::= SEQUENCE {
    teaPwPeerMappingPeerAddrType      InetAddressType,
    teaPwPeerMappingPeerAddr          InetAddress,
    teaPwPeerMappingPwType            TeaPwTypeTC,
    teaPwPeerMappingPwID              TeaPwIDType,
    teaPwPeerMappingPwIndex           TeaPwIndexType
}

teaPwPeerMappingPeerAddrType OBJECT-TYPE
    SYNTAX          InetAddressType
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "IP address type of the peer node."
    ::= { teaPwPeerMappingEntry 1 }

teaPwPeerMappingPeerAddr OBJECT-TYPE
    SYNTAX          InetAddress
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "IP address type of the peer node."
    ::= { teaPwPeerMappingEntry 2 }

teaPwPeerMappingPwType OBJECT-TYPE
    SYNTAX          TeaPwTypeTC
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The PW type (indicates the service) of this PW."
    ::= { teaPwPeerMappingEntry 3 }

teaPwPeerMappingPwID OBJECT-TYPE
    SYNTAX          TeaPwIDType
    MAX-ACCESS      not-accessible
    STATUS          current

```

```

DESCRIPTION
    "The PW ID of this PW. Zero if the PW is configured
    manually."
 ::= { teaPwPeerMappingEntry 4 }

teaPwPeerMappingPwIndex OBJECT-TYPE
    SYNTAX      TeaPwIndexType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value that represents the PW in the teaPwTable."
 ::= { teaPwPeerMappingEntry 5 }

-- End of the peer mapping table

-- End of reverse mapping tables

teaPwUpDownNotifEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "If this object is set to true(1), then it enables
        the emission of pwUp and pwDown
        notifications; otherwise these notifications are not
        emitted."
    REFERENCE
        "See also [RFC 3413] for explanation that
        notifications are under the ultimate control of the
        MIB module in this document."
    DEFVAL { false }
 ::= { teaPwObjects 9 }

teaPwDeletedNotifEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "If this object is set to true(1), then it enables the
        emission of teaPwDeleted notification; otherwise this
        notification is not emitted."
    REFERENCE
        "See also [RFC 3413] for explanation that
        notifications are under the ultimate control of the
        MIB module in this document."
    DEFVAL { false }
 ::= { teaPwObjects 10 }

teaPwNotifRate OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "This object defines the maximum number of PW notifications
        that can be emitted from the device per second."
 ::= { teaPwObjects 11 }

-- Notifications - PW

teaPwDown NOTIFICATION-TYPE
    OBJECTS { teaPwOperStatus, --start of range
             teaPwOperStatus --end of range
            }
    STATUS      current
    DESCRIPTION
        "This notification is generated when the
        teaPwOperStatus object for one or more contiguous
        entries in teaPwTable are about to enter the
        down(2) state from some other state. The included values
        of teaPwOperStatus MUST all be set equal to this

```

```

down(2) state. The two instances of teaPwOperStatus
in this notification indicate the range of indexes
that are affected. Note that all the indexes of the
two ends of the range can be derived from the
instance identifiers of these two objects. For
cases where a contiguous range of cross-connects
have transitioned into the down(2) state at roughly
the same time, the device SHOULD issue a single
notification for each range of contiguous indexes in
an effort to minimize the emission of a large number
of notifications. If a notification has to be
issued for just a single cross-connect entry, then
the instance identifier (and values) of the two
teaPwOperStatus objects MUST be identical."
 ::= { teaPwNotifications 1 }

teaPwUp NOTIFICATION-TYPE
OBJECTS { teaPwOperStatus, --start of range
          teaPwOperStatus --end of range
        }
STATUS current
DESCRIPTION
    "This notification is generated when the
    teaPwOperStatus object for one or more contiguous
    entries in teaPwTable are about to enter the up(1)
    state from some other state. The included values of
    teaPwOperStatus MUST both be set equal to this
    new state (i.e., up(1)). The two instances of
    teaPwOperStatus in this notification indicate the range
    of indexes that are affected. Note that all the indexes
    of the two ends of the range can be derived from the
    instance identifiers of these two objects. For
    cases where a contiguous range of cross-connects
    have transitioned into the up(1) state at roughly
    the same time, the device SHOULD issue a single
    notification for each range of contiguous indexes in
    an effort to minimize the emission of a large number
    of notifications. If a notification has to be
    issued for just a single cross-connect entry, then
    the instance identifier (and values) of the two
    teaPwOperStatus objects MUST be identical."
 ::= { teaPwNotifications 2 }

teaPwDeleted NOTIFICATION-TYPE
OBJECTS { teaPwType,
          teaPwID,
          teaPwPeerAddrType,
          teaPwPeerAddr
        }
STATUS current
DESCRIPTION
    "This notification is generated when the PW has been
    deleted, i.e., when the teaPwRowStatus has been set to
    destroy(6), the PW has been deleted by a non-MIB
    application or due to auto discovery process.
    "
 ::= { teaPwNotifications 3 }

-- End of notifications.

-- conformance information

teaPwGroups      OBJECT IDENTIFIER ::= { teaPwConformance 1 }
teaPwCompliances OBJECT IDENTIFIER ::= { teaPwConformance 2 }

-- Compliance requirement for fully compliant implementations.

pwModuleFullCompliance MODULE-COMPLIANCE
STATUS current

```

```

DESCRIPTION
    "The compliance statement for agents that provide full
    support for PW MIB Module. Such devices can
    then be monitored and also be configured using
    this MIB module."

MODULE -- this module
    MANDATORY-GROUPS { teaPwBasicGroup,
                       teaPwPerformanceGeneralGroup
                     }
 ::= { teaPwCompliances 1 }

bsodPwModuleFullCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "The compliance statement for the PW MIB module
        that supports TDM Emulation according to the
        Business Service over Docsis (BSOD) specification
        published by CableLabs.
        "

MODULE -- this module
    MANDATORY-GROUPS { bsodPwBasicGroup,
                       teaPwPerformanceGeneralGroup,
                       teaPwPerformanceIntervalGroup
                     }

GROUP teaPwNotificationGroup
DESCRIPTION "This group is only mandatory for those
            implementations
            which can efficiently implement the notifications
            contained in this group."

GROUP      teaPwPwIdGroup
DESCRIPTION "This group is only mandatory for implementations
            that support the PW ID FEC.
            "

GROUP      teaPwGeneralizedFecGroup
DESCRIPTION "This group is only mandatory for implementations
            that support the generalized PW FEC.
            "

GROUP      teaPwFcsGroup
DESCRIPTION "This group is only mandatory for implementations
            that support FCS retention."

GROUP      teaPwFragGroup
DESCRIPTION "This group is only mandatory for implementations
            that support PW fragmentation.
            "

GROUP      teaPwPwStatusGroup
DESCRIPTION "This group is only mandatory for implementations
            that support PW status notification.
            "

GROUP      teaPwGetNextGroup
DESCRIPTION "This group is only mandatory for implementations
            where the teaPwIndex may be any arbitrary value
            and the EMS would require retrieval of the next
            free index."

GROUP      teaPwPriorityGroup
DESCRIPTION "This group is only mandatory for implementations
            that support controlling the PW setup priority and the PW
            holding priority."

GROUP      teaPwAttachmentGroup

```

```

DESCRIPTION "This group is only mandatory for implementations
that support attachment of two PWs (PW stitching)."
```

```

GROUP      teaPwPerformance1DayIntervalGroup
DESCRIPTION "This group is only mandatory for implementations
that support PW performance gathering in 1-day
intervals."
```

```

GROUP      teaPwPerformanceIntervalGeneralGroup
DESCRIPTION "This group is only mandatory for implementations
that support PW performance gathering in 15-minute
intervals."
```

```

GROUP      teaPwPerformanceIntervalGroup
DESCRIPTION "This group is only mandatory for implementations
that support PW performance gathering in 15-minute
intervals."
```

```

GROUP      teaPwHCPeformanceIntervalGroup
DESCRIPTION "This group is only mandatory for implementations
where at least one of the interval performance
counters wraps around too quickly based on the
criteria specified in RFC 2863 for high-capacity
counters."
```

```

GROUP      teaPwMappingTablesGroup
DESCRIPTION "This group is only mandatory for implementations
that support reverse mapping of PW indexes to
the teaPwIndex and the peer mapping table."
```

```

GROUP      teaPwNotificationControlGroup
DESCRIPTION "This group is only mandatory for implementations
that support the PW notifications."
```

```

OBJECT     teaPwAdminStatus
SYNTAX     INTEGER { up(1), down(2) }
DESCRIPTION "The support of the value testing(3) is not
required."
```

```

OBJECT     teaPwOperStatus
SYNTAX     INTEGER { up(1), down(2), notPresent(6),
lowerLayerDown(7) }
DESCRIPTION "The support of the values testing(3), unknown(4),
and dormant(5) is not required."
```

```

OBJECT     teaPwRowStatus
SYNTAX     RowStatus { active(1), notInService(2),
notReady(3) }
WRITE-SYNTAX RowStatus { active(1), notInService(2),
createAndGo(4), destroy(6)
}
DESCRIPTION "Support for createAndWait is not required. Support
of notReady is not required for implementations
that do not support signaling, or if it is
guaranteed that the conceptual row has all the
required information to create the PW when the
row has been created by the agent."
```

```

OBJECT     teaPwPeerAddrType
SYNTAX     InetAddressType { unknown(0), ipv4(1) }
DESCRIPTION "Only unknown(0) and ipv4(1) is required."
```

```

OBJECT     teaPwPeerAddr
SYNTAX     InetAddress (SIZE(0|4))
DESCRIPTION "An implementation is only required to support
0, 4 address sizes."
```

```

 ::= { teaPwCompliances 2 }

-- Compliance requirement for read-only compliant implementations.
```

```

teaPwModuleReadOnlyCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "The compliance statement for agents that provide read-only
    support for PW MIB Module. Such devices can then be
    monitored but cannot be configured using this
    MIB module."

  MODULE -- this module
    MANDATORY-GROUPS { teaPwBasicGroup
                      }

  GROUP      teaPwNotificationGroup
  DESCRIPTION "This group is only mandatory for those
  implementations
  which can efficiently implement the notifications
  contained in this group."

  GROUP      teaPwPwIdGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support the PW ID FEC.
  "

  GROUP      teaPwGeneralizedFecGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support the generalized PW FEC.
  "

  GROUP      teaPwFcsGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support FCS retention."

  GROUP      teaPwFragGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support PW fragmentation.
  "

  GROUP      teaPwPwStatusGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support PW status notification.
  "

  GROUP      teaPwGetNextGroup
  DESCRIPTION "This group is only mandatory for implementations
  where the teaPwIndex may be any arbitrary value
  and the EMS would require retrieval of the next
  free index."

  GROUP      teaPwPriorityGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support controlling the PW setup priority and the
  PW holding priority."

  GROUP      teaPwAttachmentGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support attachment of two PWs (PW stitching)."
```

```

  GROUP      teaPwPerformance1DayIntervalGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support PW performance gathering in 1-day
  intervals."

  GROUP      teaPwPerformanceIntervalGeneralGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support PW performance gathering in 15-minute
  intervals."

  GROUP      teaPwPerformanceIntervalGroup
  DESCRIPTION "This group is only mandatory for implementations
  that support PW performance gathering in 15-minute
  intervals."

```

GROUP teaPwHCPerformanceIntervalGroup  
DESCRIPTION "This group is only mandatory for implementations where at least one of the interval performance counters wraps around too quickly based on the criteria specified in RFC 2863 for high-capacity counters."

GROUP teaPwMappingTablesGroup  
DESCRIPTION "This group is only mandatory for implementations that support reverse mapping of PW indexes to the teaPwIndex and the peer mapping table."

GROUP teaPwNotificationControlGroup  
DESCRIPTION "This group is only mandatory for implementations that support the PW notifications."

OBJECT teaPwType  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwOwner  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwPsnType  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwSetUpPriority  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwHoldingPriority  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwPeerAddrType  
SYNTAX InetAddressType { unknown(0), ipv4(1) }  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required. Only unknown(0) and ipv4(1) is required."

OBJECT teaPwPeerAddr  
SYNTAX InetAddress (SIZE(0|4))  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required. An implementation is only required to support 0, 4 address sizes."

OBJECT teaPwAttachedPwIndex  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwIfIndex  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwID  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwLocalGroupID  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwGroupAttachmentID  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwLocalAttachmentID  
MIN-ACCESS read-only

DESCRIPTION "Write access is not required."

OBJECT teaPwPeerAttachmentID  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwCwPreference  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwLocalIfMtu  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwLocalIfString  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwLocalCapabAdvert  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwFragmentCfgSize  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwFcsRetentioncfg  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwOutboundLabel  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwInboundLabel  
MIN-ACCESS read-only

DESCRIPTION "Write access is not required."

OBJECT teaPwName  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwDescr  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwAdminStatus  
SYNTAX INTEGER { up(1), down(2) }  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required. The support of value testing(3) is not required."

OBJECT teaPwOperStatus  
SYNTAX INTEGER { up(1), down(2), notPresent(6), lowerLayerDown(7) }  
DESCRIPTION "The support of the values testing(3), unknown(4), and is not required."

OBJECT teaPwRowStatus  
SYNTAX RowStatus { active(1) }  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwStorageType  
MIN-ACCESS read-only  
DESCRIPTION "Write access is not required."

OBJECT teaPwUpDownNotifEnable  
MIN-ACCESS read-only

```

DESCRIPTION "Write access is not required."

OBJECT      teaPwDeletedNotifEnable
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      teaPwNotifRate
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

    ::= { teaPwCompliances 2 }

-- Units of conformance.

teaPwBasicGroup  OBJECT-GROUP
OBJECTS {
    teaPwType,
    teaPwOwner,
    teaPwPsnType,
    teaPwPeerAddrType,
    teaPwPeerAddr,
    teaPwIfIndex,
    teaPwID,
    teaPwLocalGroupID,
    teaPwCwPreference,
    teaPwLocalIfMtu,
    teaPwLocalIfString,
    teaPwLocalCapabAdvert,
    teaPwRemoteGroupID,
    teaPwCwStatus,
    teaPwRemoteIfMtu,
    teaPwRemoteIfString,
    teaPwOutboundLabel,
    teaPwInboundLabel,
    teaPwName,
    teaPwDescr,
    teaPwCreateTime,
    teaPwUpTime,
    teaPwLastChange,
    teaPwAdminStatus,
    teaPwOperStatus,
    teaPwLocalStatus,
    teaPwRowStatus,
    teaPwStorageType
}

STATUS current
DESCRIPTION
    "Collection of objects that are required in all
    implementations that support the PW MIB module."
    ::= { teaPwGroups 1 }

teaPwPwIdGroup  OBJECT-GROUP
OBJECTS {
    teaPwID
}

STATUS current
DESCRIPTION
    "Collection of objects required for PW ID configuration
    and signaling."
    ::= { teaPwGroups 2 }

teaPwGeneralizedFecGroup  OBJECT-GROUP
OBJECTS {
    teaPwGroupAttachmentID,
    teaPwLocalAttachmentID,
    teaPwPeerAttachmentID
}

STATUS current

```

```

DESCRIPTION
    "Collection of objects required for generalized FEC
    configuration and signaling."
 ::= { teaPwGroups 3 }

teaPwFcsGroup    OBJECT-GROUP
OBJECTS {
    teaPwFcsRetentioncfg,
    teaPwFcsRetentionStatus
}

STATUS    current
DESCRIPTION
    "Collection of objects required for FCS retention
    configuration and signaling."
 ::= { teaPwGroups 4 }

teaPwFragGroup    OBJECT-GROUP
OBJECTS {
    teaPwFragmentCfgSize,
    teaPwRmtFragCapability
}

STATUS    current
DESCRIPTION
    "Collection of objects required for fragmentation
    configuration and signaling."
 ::= { teaPwGroups 5 }

teaPwPwStatusGroup    OBJECT-GROUP
OBJECTS {
    teaPwRemoteCapabilities,
    teaPwRemoteStatusCapable,
    teaPwRemoteStatus
}

STATUS    current
DESCRIPTION
    "Collection of objects required for PW status configuration
    and signaling."
 ::= { teaPwGroups 6 }

teaPwGetNextGroup    OBJECT-GROUP
OBJECTS {
    teaPwIndexNext
}

STATUS    current
DESCRIPTION
    "Collection of objects for getting the next available
    index."
 ::= { teaPwGroups 7 }

teaPwPriorityGroup    OBJECT-GROUP
OBJECTS {
    teaPwSetUpPriority,
    teaPwHoldingPriority
}

STATUS    current
DESCRIPTION
    "Collection of objects for controlling the PW setup and
    holding priority."
 ::= { teaPwGroups 8 }

teaPwAttachmentGroup    OBJECT-GROUP
OBJECTS {
    teaPwAttachedPwIndex
}

```

```

STATUS current
DESCRIPTION
    "Collection of objects for PW configuration
    and signaling."
::= { teaPwGroups 9 }

teaPwPerformanceGeneralGroup OBJECT-GROUP
OBJECTS {
    teaPwPerfTotalErrorPackets
}

STATUS current
DESCRIPTION
    "Collection of general objects needed for managing the
    total running performance parameters."
::= { teaPwGroups 10 }

teaPwPerformance1DayIntervalGroup OBJECT-GROUP
OBJECTS {
    teaPwPerf1DayIntervalValidData,
    teaPwPerf1DayIntervalMoniSecs,
    teaPwPerf1DayIntervalInHCPackets,
    teaPwPerf1DayIntervalInHCBytes,
    teaPwPerf1DayIntervalOutHCPackets,
    teaPwPerf1DayIntervalOutHCBytes
}

STATUS current
DESCRIPTION
    "Collection of objects needed for PW running 1-day interval
    performance collection."
::= { teaPwGroups 11 }

teaPwPerformanceIntervalGeneralGroup OBJECT-GROUP
OBJECTS {
    teaPwTimeElapsed,
    teaPwValidIntervals,
    teaPwPerfIntervalValidData,
    teaPwPerfIntervalTimeElapsed
}

STATUS current
DESCRIPTION
    "Collection of general objects needed for managing the
    interval performance parameters."
::= { teaPwGroups 12 }

teaPwPerformanceIntervalGroup OBJECT-GROUP
OBJECTS {
    teaPwPerfCurrentInPackets,
    teaPwPerfCurrentInBytes,
    teaPwPerfCurrentOutPackets,
    teaPwPerfCurrentOutBytes,

    teaPwPerfIntervalInPackets,
    teaPwPerfIntervalInBytes,
    teaPwPerfIntervalOutPackets,
    teaPwPerfIntervalOutBytes
}

STATUS current
DESCRIPTION
    "Collection of 32 bits objects needed for PW performance
    collection in 15-minute intervals."
::= { teaPwGroups 13 }

teaPwHCPeformanceIntervalGroup OBJECT-GROUP
OBJECTS {
    teaPwPerfCurrentInHCPackets,
    teaPwPerfCurrentInHCBytes,
    teaPwPerfCurrentOutHCPackets,

```

```

        teaPwPerfCurrentOutHCBytes,

        teaPwPerfIntervalInHCPackets,
        teaPwPerfIntervalInHCBytes,
        teaPwPerfIntervalOutHCPackets,
        teaPwPerfIntervalOutHCBytes
    }

STATUS current
DESCRIPTION
    "Collection of HC objects needed for PW performance
    collection in 15-minute intervals."
::= { teaPwGroups 14 }

teaPwMappingTablesGroup OBJECT-GROUP
OBJECTS {
    teaPwIndexMappingPwIndex,
    teaPwPeerMappingPwIndex
}

STATUS current
DESCRIPTION
    "Collection of objects contained in the reverse
    mapping tables."
::= { teaPwGroups 15 }

teaPwNotificationControlGroup OBJECT-GROUP
OBJECTS {
    teaPwUpDownNotifEnable,
    teaPwDeletedNotifEnable,
    teaPwNotifRate
}

STATUS current
DESCRIPTION
    "Collection of objects for controlling the PW
    notifications."
::= { teaPwGroups 16 }

teaPwNotificationGroup NOTIFICATION-GROUP
NOTIFICATIONS {
    teaPwUp,
    teaPwDown,
    teaPwDeleted
}

STATUS current
DESCRIPTION
    "Collection of PW notifications objects."
::= { teaPwGroups 17 }

bsodPwBasicGroup OBJECT-GROUP
OBJECTS {
    teaPwType,
    --teaPwOwner,
    teaPwPsnType,
    teaPwPeerAddrType,
    teaPwPeerAddr,
    teaPwIfIndex,
    --teaPwID,
    --teaPwLocalGroupID,
    --teaPwCwPreference,
    --teaPwLocalIfMtu,
    --teaPwLocalIfString,
    --teaPwLocalCapabAdvert,
    --teaPwRemoteGroupID,
    --teaPwCwStatus,
    --teaPwRemoteIfMtu,
    --teaPwRemoteIfString,
    teaPwOutboundLabel,
    teaPwInboundLabel,
}

```

```

        teaPwName,
        teaPwDescr,
        teaPwCreateTime,
        teaPwUpTime,
        teaPwLastChange,
        teaPwAdminStatus,
        teaPwOperStatus,
        teaPwLocalStatus,
        teaPwRowStatus
        --teaPwStorageType
    }

    STATUS current
    DESCRIPTION
        "Collection of objects that are required in the BSoD PW MIB module."
    ::= { teaPwGroups 18 }

END

```

### B.3 CL-PW-TDM-MIB

```

CL-PW-TDM-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE,
    Integer32, Counter32, Unsigned32
        FROM SNMPv2-SMI

    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF

    TEXTUAL-CONVENTION, TruthValue, RowStatus, StorageType,
    TimeStamp, TestAndIncr
        FROM SNMPv2-TC

    InterfaceIndexOrZero
        FROM IF-MIB

    PerfCurrentCount, PerfIntervalCount
        FROM PerfHist-TC-MIB

    HCPerfTimeElapsed
        FROM HC-PerfHist-TC-MIB          -- [RFC 3705]

    teaPwIndex
        FROM CL-PW-MIB

    clabProjDocsis
        FROM CLAB-DEF-MIB;

-- The TDM MIB

teaPwTDM MIB MODULE-IDENTITY
LAST-UPDATED "200602060000Z" -- February 6, 2006
ORGANIZATION "Cable Television Laboratories, Inc"
CONTACT-INFO
    "
        Postal: Cable Television Laboratories, Inc.
        858 Coal Creek Circle
        Louisville, Colorado 80027-9750
        U.S.A.
        Phone: +1 303-661-9100
        Fax: +1 303-661-9199
        E-mail: mibs@cablelabs.com"
DESCRIPTION
    "
        This MIB contains managed object definitions for
        encapsulating TDM (T1, E1, T3, E3, NxDS0) as
        pseudo-wires over packet-switching networks (PSN).
    "

```

This MIB supplements the PW-STD-MIB as in: Zelig, D., Nadeau, T. 'Pseudo Wire (PW) Management Information Base'. The PW-STD-MIB contains structures and MIB associations generic to Pseudo-Wire (PW) emulation. PW-specific MIBs (such as this) contain config and stats for specific PW types.

Initial version, published as draft-ietf-pwe3-TDM-mib-00.txt by the 'Pseudo-Wire Emulation Edge-to-Edge (PWE3) Working Group'

For more info, see the PWE3 Working Group (email distribution pwe3@ietf.org, <http://www.ietf.org/html.charters/pwe3-charter.html>)

This version was taken from a snapshot of draft-ietf-pwe3-pw-tdm-mib-04.txt (by Orly Nicklass as noted below) and modified by David Brief and Ron Cohen of Resolute Networks to be rooted in the under DOCSIS projects in the CableLabs private enterprises branch. The intention is to support the standard MIBs when they reach RFC status.  
"

REVISION

"200602060000Z" -- February 6, 2006

DESCRIPTION

"This revision, published as CL-PW-TDM-MIB. Changes from ietf-pw-tdm-mib-04  
-add in tea prefix to avoid problems when migrating to RFC  
  
-add RTP SSRC variable per TDM pw  
  
    teaPwTDMRtpSSRC  
    teaPwTDMPeerRtpSSRC  
  
-add additional objects for configuring policy of IWF:  
    teaPwTDMCfgFillerPattern  
    teaPwTDMCfgLflagPayloadPolicy  
  
-add additional objects for configuring TOS of packets:  
    teaPwTDMCfgIPTos  
  
-add additional objects for RTP configuration  
    teaPwTDMCfgRtpPT  
    teaPwTDMCfgPeerRtpPT  
    teaPwTDMCfgRtpTSRef  
    teaPwTDMCfgPeerRtpTSRef  
    teaPwTDMCfgSRTPenable  
  
-add in additional table teaPwTDMCfgFramedTable for configuration of CESoPSN specific information  
  
-add in additional objects for configuring policies:  
  
    teaPwTDMCfgFramedIdlePattern  
    teaPwTDMCfgFramedRflagPolicy  
    teaPwTDMCfgFramedLflagPolicy  
    teaPwTDMCfgFramedRDPolicy  
    teaPwTDMCfgFramedLopsPolicy  
  
-add in additional objects for configuring signaling mode:  
  
    teaPwTDMCfgFramedSigPT  
    teaPwTDMCfgFramedSigPeerPT  
    teaPwTDMCfgFramedSigIPTos  
    teaPwTDMCfgFramedSigIdle  
    teaPwTDMCfgFramedSigInterval  
    teaPwTDMCfgFramedSigMaxInterval

```

"
-- pwVcTDM MIB MODULE-IDENTITY
--
-- LAST-UPDATED "200510230000Z"
-- ORGANIZATION "Pseudo-Wire Emulation Edge-to-Edge (PWE3)
-- Working Group"
-- CONTACT-INFO
-- " Orly Nicklass
-- Postal: RAD Data Communications
-- 24 Raoul Wallenberg St., Bldg C
-- Tel Aviv 69719, Israel
-- Email: orly_n@rad.com
--
-- The PWE3 Working Group (email distribution pwe3@ietf.org,
-- http://www.ietf.org/html.charters/pwe3-charter.html)
-- "
--
-- DESCRIPTION
-- "This MIB contains managed object definitions for
-- encapsulating TDM (T1,E1, T3, E3, NxDS0) as
-- pseudo-wires over packet-switching networks (PSN).
--
-- This MIB supplements the PW-STD-MIB as in: Zelig, D.,
-- Nadeau, T. 'Pseudo Wire (PW) Management Information Base'.
-- The PW-STD-MIB contains structures and MIB associations
-- generic to Pseudo-Wire (PW) emulation. PW-specific
-- MIBs (such as this) contain config and stats for specific
-- PW types.
--
-- Copyright (C) The Internet Society (2005). This version
-- of this MIB module is part of RFC yyyy; see the RFC
-- itself for full legal notices.
-- RFC Ed.: replace yyyy with actual RFC number & remove this
-- note"

 ::= { clabProjDocsis 16 }
 -- ::= { pwStdMIB XXX }

-- Local Textual conventions

TeaPwTDMCfgIndex ::= TEXTUAL-CONVENTION
  STATUS current
  DESCRIPTION
    "Index into any of the relevant teaPwXXXXCfTable."
  SYNTAX Unsigned32

-- Tables, Scalars
teaPwTDMObjects OBJECT IDENTIFIER
  ::= { teaPwTDM MIB 1 }

-- Notifications
teaPwTDMTraps OBJECT IDENTIFIER
  ::= { teaPwTDM MIB 2 }

-- Conformance
teaPwTDMConformance OBJECT IDENTIFIER
  ::= { teaPwTDM MIB 3 }

-- TDM PW table

teaPwTDMTable OBJECT-TYPE
  SYNTAX SEQUENCE OF TeaPwTDMEntry
  MAX-ACCESS not-accessible
  STATUS current
  DESCRIPTION
    "This table contains basic information including ifIndex,
    and pointers to entries in the relevant TDM config

```

```

        tables for this TDM PW."
 ::= { teaPwTDMObjects 1 }

teaPwTDMEntry OBJECT-TYPE
    SYNTAX      TeaPwTDMEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table is indexed by the same index that was
        created for the associated entry in the PW Table
        (in the TEA-PW-MIB).

        - The teaPwIndex.

        An entry is created in this table by the agent for every
        entry in the teaPwTable with a teaPwType equal to one of the
        following:
        e1Satop(17), t1Satop(18), e3Satop(19), t3Satop(20),
        basicCesPsn(21), basicTdmIp(22), tdmCasCesPsn(23),
        tdmCasTdmIp(24)."
```

```

INDEX { teaPwIndex }

 ::= { teaPwTDMTable 1 }

TeaPwTDMEntry ::= SEQUENCE {

    teaPwTDMRate                Integer32,
    teaPwTDMIfIndex             InterfaceIndexOrZero,
    teaPwGenTDMCfgIndex         TeaPwTDMCfgIndex,
    teaPwRelTDMCfgIndex         TeaPwTDMCfgIndex,

    teaPwTDMConfigError         BITS,
    teaPwTDMTimeElapsed         Integer32,
    teaPwTDMValidIntervals     Integer32,
    teaPwTDMCurrentIndications  BITS,
    teaPwTDMlatchedIndications  BITS,
    teaPwTDMLastEsTimeStamp     TimeStamp,

    teaPwTDMRtpSSRC             Unsigned32,
    teaPwTDMPeerRtpSSRC         Unsigned32
}

teaPwTDMRate OBJECT-TYPE
    SYNTAX      Integer32
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The parameter represents the bit-rate of the TDM service
        in multiples of the 'basic' 64 kbit/s rate. It complements
        the definition of pwType used in PW-STD-MIB.
        For structure-agnostic the following should be used:
        a) Satop E1 - 32
        b) Satop T1 emulation:
            i) MUST be set to 24 in the basic emulation mode
            ii) MUST be set to 25 for the 'Octet-aligned T1'
                emulation mode
        c) Satop E3 - 535
        d) Satop T3 - 699
        For all kinds of structure-aware emulation, this parameter
        MUST be set to N where N is the number of DS0 channels
        in the corresponding attachment circuit."
    REFERENCE
        "See [TDMCP-EXT]"
    DEFVAL { 32 }
 ::= { teaPwTDMEntry 1 }

teaPwTDMIfIndex OBJECT-TYPE
    SYNTAX      InterfaceIndexOrZero
    MAX-ACCESS  read-write

```

```

STATUS          current
DESCRIPTION
    "This is a unique index within the ifTable. It represents
    the interface index of the full link or the interface
    index for the bundle holding the group of
    time slots to be transmitted via this PW connection.

    A value of zero indicates an interface index that has yet
    to be determined.
    Once set, if the TDM ifIndex is (for some reason) later
    removed, the agent SHOULD delete the associated PW rows
    (e.g., this teaPwTDMTable entry). If the agent does not
    delete the rows, the agent MUST set this object to
    zero."
 ::= { teaPwTDMEntry 2 }

teaPwGenTDMCfgIndex OBJECT-TYPE
SYNTAX          TeaPwTDMCfgIndex
MAX-ACCESS      read-write
STATUS          current
DESCRIPTION
    "Index to the generic parameters in the TDM configuration
    table that appears in this MIB module. It is likely that
    multiple TDM PWs of the same characteristic will share
    a single TDM Cfg entry."
 ::= { teaPwTDMEntry 3 }

teaPwRelTDMCfgIndex OBJECT-TYPE
SYNTAX          TeaPwTDMCfgIndex
MAX-ACCESS      read-write
STATUS          current
DESCRIPTION
    "Index to the relevant TDM configuration table entry
    that appears in one of the related MIB modules
    such as TDMoIP or CESoPSN. It is likely that
    multiple TDM PWs of the same characteristic will share
    a single configuration entry of the relevant type.
    The value 0 implies no entry in other related MIBs.

    In this version of the MIB, this pointer points to
    An entry in teaPwTDMCfgFramedTable for CESoPSN TDM
    PWs. For other PW types, it is set to none."

 ::= { teaPwTDMEntry 4 }

teaPwTDMConfigError OBJECT-TYPE
SYNTAX BITS {
    other ( 0),
    tdmTypeIncompatible ( 1),
    peerRtpIncompatible ( 2),
    peerPayloadSizeIncompatible ( 3)
}
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Any of the bits are set if the local configuration is
    not compatible with the peer configuration as available
    from the various parameters options.

    -tdmTypeIncompatible bit is set if the local configuration
    is not carrying the same TDM type as the peer configuration.

    -peerRtpIncompatible bit is set if the local configuration
    is configured to send RTP packets for this PW, and the
    remote is not capable of accepting RTP packets.

    -peerPayloadSizeIncompatible bit is set if the local
    configuration is not carrying the same Payload Size as the
    peer configuration.  "
 ::= { teaPwTDMEntry 5}

```

teaPwTDMTimeElapsed OBJECT-TYPE

SYNTAX Integer32 (1..900)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of seconds, including partial seconds, that have elapsed since the beginning of the current measurement period. If, for some reason, such as an adjustment in the system's time-of-day clock, the current interval exceeds the maximum value, the agent will return the maximum value."

::= { teaPwTDMEntry 6 }

teaPwTDMValidIntervals OBJECT-TYPE

SYNTAX Integer32 (0..96)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of previous 15-minute intervals for which data was collected.

An agent with TDM capability must be capable of supporting at least n intervals. The minimum value of n is 4, The default of n is 32 and the maximum value of n is 96.

The value will be <n> unless the measurement was (re-) started within the last (<n>\*15) minutes, in which case the value will be the number of complete 15-minute intervals for which the agent has at least some data.

In certain cases(e.g., in the case where the agent is a proxy) it is possible that some intervals are unavailable. In this case, this interval is the maximum interval number for which data is available. "

::= { teaPwTDMEntry 7 }

teaPwTDMCurrentIndications OBJECT-TYPE

SYNTAX BITS {

other ( 0 ),

strayPacket ( 1 ),

malformedPacket ( 2 ),

excessivePktLossRate( 3 ),

bufferOverrun ( 4 ),

bufferUnderrun ( 5 ),

remotePktLoss ( 6 ),

pktMisOrder ( 7 ),

packetLoss ( 8 ),

tdmFault ( 9 )

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The following defects should be detected and reported upon request:

-Stray packets MAY be detected by the PSN and multiplexing layers. Stray packets MUST be discarded by the CE-bound IWF and their detection MUST NOT affect mechanisms for detection of packet loss.

-Malformed packets are detected by mismatch between the expected packet size (taking the value of the L bit into account) and the actual packet size inferred from the PSN and multiplexing layers. Malformed in-order packets MUST be discarded by the CE-bound IWF and replacement data generated

as for lost packets.

-Excessive packet loss rate is detected by computing the average packet loss rate over the value of teaPwTDMavePktLossTimeWindow and comparing it with a preconfigured threshold [RFC 4553].

-Buffer overrun is detected in the normal operation state when the CE bound IWF's jitter buffer cannot accommodate newly arrived packets.

-Remote packet loss is indicated by reception of packets with their R bit set.

-Packet misorder is detected by looking at the Sequence number provided by the control word.

-TDM Fault, if L bit in the control word is set, it indicates that TDM data carried in the payload is invalid due an attachment circuit fault. When the L bit is set the payload MAY be omitted in order to conserve bandwidth.

Note: the algorithm used to capture these indications is implementation-specific."

```
::= { teaPwTDMEntry 8}
```

teaPwTDM LatchedIndications OBJECT-TYPE

```
SYNTAX BITS {
    other                ( 0),
    staryPacket          ( 1),
    malformedPacket      ( 2),
    excessivePktLossRate( 3),
    bufferOverrun        ( 4),
    bufferUnderrun       ( 5),
    remotePktLoss        ( 6),
    pktMisOrder          ( 7),
    packetLoss           ( 8),
    tdmFault              ( 9)
```

```
}
```

```
MAX-ACCESS    read-only
```

```
STATUS        current
```

```
DESCRIPTION
```

```
"The state of TDM indicators when the TDM PW last declared an error second (either as ES, SES or a second with errors inside a UAS) condition. At this time, only LOPS can create a failure. Since indicators other than LOPS are useful, all are latched here. For bit definitions, see teaPwTDMCurrentIndications above.
```

Note: the algorithm used to latch these indications when entering a defect state is implementation-specific."

```
::= { teaPwTDMEntry 9}
```

teaPwTDM LastEsTimeStamp OBJECT-TYPE

```
SYNTAX        TimeStamp
```

```
MAX-ACCESS    read-only
```

```
STATUS        current
```

```
DESCRIPTION
```

```
"The value of sysUpTime at the most recent occasion at which the TDM PW entered the ES or SES state."
```

```
::= { teaPwTDMEntry 10}
```

teaPwTDM RtpSSRC OBJECT-TYPE

```
SYNTAX        Unsigned32
```

```
MAX-ACCESS    read-create
```

```
STATUS        current
```

```
DESCRIPTION
```

```
"In the case where RTP headers are being used on this PW, this object configures the value of the SSRC field to use in the RTP header "
```

```
::= { teaPwTDMEntry 11}
```

```

teaPwTDMPeerRtpSSRC OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "In the case where RTP headers are being used on this PW,
    this object configures the value of the SSRC field to
    expect in the RTP header.  When this value is non-zero
    it may be used to check incoming data frames for validity.  "
DEFVAL { 0 }

 ::= { teaPwTDMEntry 12}

-- End of TDM PW table

-- PW Generic TDM PW Configuration Table

teaPwTDMCfgIndexNext OBJECT-TYPE
SYNTAX      TestAndIncr
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This object contains the value to be used for
    teaPwTDMCfgIndex when creating entries in the
    teaPwTDMCfgTable.  The value 0 indicates that no
    unassigned entries are available.  To obtain the
    value of teaPwTDMCfgIndexNext for a new entry in the
    teaPwTDMCfgTable, the manager issues a management
    protocol retrieval operation to obtain the current
    value of teaPwTDMCfgIndexNext.  After each retrieval
    operation, the agent should modify the value to
    reflect the next unassigned index.  After a manager
    retrieves a value the agent will determine through
    its local policy when this index value will be made
    available for reuse."
 ::= { teaPwTDMObjects 2 }

teaPwTDMCfgTable OBJECT-TYPE
SYNTAX      SEQUENCE OF TeaPwTDMCfgEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "This table contains a set of parameters that may be
    referenced by one or more TDM PWs in teaPwTDMTable."

 ::= { teaPwTDMObjects 3 }

teaPwTDMCfgEntry OBJECT-TYPE
SYNTAX      TeaPwTDMCfgEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "These parameters define the characteristics of a
    TDM PW.  They are grouped here to ease NMS burden.
    Once an entry is created here it may be reused
    by many PWs."

INDEX { teaPwGenTDMCfgIndex }

 ::= { teaPwTDMCfgTable 1 }

TeaPwTDMCfgEntry ::= SEQUENCE {
    teaPwTDMCfgIndex          TeaPwTDMCfgIndex,
    teaPwTDMCfgRowStatus     RowStatus,
    teaPwTDMCfgConfErr       BITS,
    teaPwTDMCfgPayloadSize   Unsigned32,
    teaPwTDMCfgPktReorder    TruthValue,

```

teaPwTDMCcfgRtpHdrUsed	TruthValue,
teaPwTDMCcfgJtrBfrDepth	Unsigned32,
teaPwTDMCcfgPayloadSuppression	INTEGER,
teaPwTDMCcfgConsecPktsInSynch	Unsigned32,
teaPwTDMCcfgConsecMissPktsOutSynch	Unsigned32,
teaPwTDMCcfgSetUp2SynchTimeOut	Unsigned32,
teaPwTDMCcfgPktReplacePolicy	INTEGER,
teaPwTDMCcfgAvePktLossTimeWindow	Integer32,
teaPwTDMCcfgExcessivePktLossThreshold	Unsigned32,
teaPwTDMCcfgAlarmThreshold	Unsigned32,
teaPwTDMCcfgClearAlarmThreshold	Unsigned32,
teaPwTDMCcfgMissingPktsToSes	Unsigned32,
teaPwTDMCcfgTimestampMode	INTEGER,
teaPwTDMCcfgStorageType	StorageType,

-- new parameters added to teaMIB and proposed to pwe3

teaPwTDMCcfgFillerPattern	Unsigned32,
teaPwTDMCcfgLflagPayloadPolicy	INTEGER,
teaPwTDMCcfgIPTos	Unsigned32,
teaPwTDMCcfgRtpPT	Unsigned32,
teaPwTDMCcfgPeerRtpPT	Unsigned32,
teaPwTDMCcfgRtpTSRef	Unsigned32,
teaPwTDMCcfgPeerRtpTSRef	Unsigned32,
teaPwTDMCcfgSRTPenable	TruthValue

}

teaPwTDMCcfgIndex OBJECT-TYPE  
 SYNTAX TeaPwTDMCcfgIndex  
 MAX-ACCESS not-accessible  
 STATUS current  
 DESCRIPTION

"Index to an entry in this table. The value is a copy of the assigned teaPwTDMCcfgIndexNext"

::= { teaPwTDMCcfgEntry 1 }

teaPwTDMCcfgRowStatus OBJECT-TYPE  
 SYNTAX RowStatus  
 MAX-ACCESS read-create  
 STATUS current  
 DESCRIPTION

"Object used for creating, modifying, and deleting a row from this table. The following objects should not be modified if the entry is in use and the status is active: teaPwTDMCcfgPayloadSize, teaPwTDMCcfgJtrBfrDepth, and teaPwTDMCcfgPayloadSuppression.

The row should not be deleted if the entry is in use"

::= { teaPwTDMCcfgEntry 2 }

teaPwTDMCcfgConfErr OBJECT-TYPE  
 SYNTAX BITS {  
     other ( 0 ),  
     payloadSize ( 1 ),  
     jtrBfrDepth ( 2 )  
 }  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION

"Various configuration errors. Illegal settings within the teaPwTDMCcfg table."

::= { teaPwTDMCcfgEntry 3 }

teaPwTDMCfPayloadSize OBJECT-TYPE  
 SYNTAX Unsigned32  
 MAX-ACCESS read-create  
 STATUS current  
 DESCRIPTION  
 "The value of this object indicates the Payload Size (in bytes) to be defined during the PW initialization. Upon TX, an implementation must be capable of carrying that amount of bytes.  
 Upon RX, when the LEN field is set to 0, the payload of packet MUST assume this size, and if the actual packet size is inconsistent with this length, the packet MUST be considered to be malformed. "  
 ::= { teaPwTDMCfEntry 4 }

teaPwTDMCfPktReorder OBJECT-TYPE  
 SYNTAX TruthValue  
 MAX-ACCESS read-create  
 STATUS current  
 DESCRIPTION  
 "If set True: as CE bound packets are queued in the jitter buffer, out of order packets are reordered. The maximum sequence number differential (i.e., the range in which re-sequencing can occur) is dependant on the depth of the jitter buffer. See teaPwTDMCfJtrBfrDepth.  
 NOTE: Some implementations may not support this feature. The agent is then required to set this to False."  
 DEFVAL { true }  
 ::= { teaPwTDMCfEntry 5 }

teaPwTDMCfRtpHdrUsed OBJECT-TYPE  
 SYNTAX TruthValue  
 MAX-ACCESS read-create  
 STATUS current  
 DESCRIPTION  
 "If set to False: an RTP header is not pre-pended to the TDM packet."  
 REFERENCE  
 "See [SATOP]"  
 DEFVAL { false }  
 ::= { teaPwTDMCfEntry 6 }

teaPwTDMCfJtrBfrDepth OBJECT-TYPE  
 SYNTAX Unsigned32  
 UNITS "microsecond"  
 MAX-ACCESS read-create  
 STATUS current  
 DESCRIPTION  
 "The size of this buffer SHOULD be locally configured to allow accommodation to the PSN-specific packet delay variation.  
 If configured to a value not supported by the implementation, the agent MUST return an error code 'jtrBfrDepth' in 'teaPwTDMConfigError '  
 NOTE: jitter buffers are a limited resource to be managed. The actual size should be at least twice as big as the value of teaPwTDMCfJtrBfrDepth "  
 DEFVAL { 3000 }  
 ::= { teaPwTDMCfEntry 7 }

teaPwTDMCfPayloadSuppression OBJECT-TYPE  
 SYNTAX INTEGER  
 {  
 enable ( 1 ),  
 disable ( 2 )

```

    }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Selecting 'enable' means: Payload suppression is allowed.
    Payload MAY be omitted in order to conserve bandwidth.
    Selecting 'disable' means: no suppression under any
    condition."
DEFVAL { disable }

 ::= { teaPwTDMCfgEntry 8 }

teaPwTDMCfgConsecPktsInSynch      OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The required number of packets with sequential
    sequence numbers that are available to be played out of the
    jitter buffer in order to exit the LOPS state.  If set to 0,
    the LOPS state is exited when the excessive packet loss rate
    falls below the excessive packet threshold value for a
    complete window time (teaPwTDMCfgAvePktLossTimeWindow)."
DEFVAL { 0 }
 ::= { teaPwTDMCfgEntry 9 }

teaPwTDMCfgConsecMissPktsOutSynch OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The number of consecutive missing packets on playout that are
    required to enter the LOPS state.  When the parameter is set to
    0, the LOPS state is entered when the excessive packet loss
    threshold has been reached."
DEFVAL { 0 }
 ::= { teaPwTDMCfgEntry 10 }

teaPwTDMCfgSetUp2SynchTimeOut OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The intermediate state timer is set to this value.  The
    Interworking Function (IWF) is in an intermediate state until
    teaPwTDMCfgConsecPktsInSynch consecutive TDM packets
    have been received or until this
    timer expires.  The timer units are (milli-sec)"

DEFVAL { 5 }
 ::= { teaPwTDMCfgEntry 11 }

teaPwTDMCfgPktReplacePolicy OBJECT-TYPE
SYNTAX          INTEGER
                {ais                (1),
                 fillerPattern       (2),
                 implementationSpecific (3)}
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    " This parameter determines the information played out
    on the TDM-bound interface instead of a missing packet
    assuming the pseudo wire is not in the LOPS state

    ais: The AIS pattern is played out

    fillerPattern: The filler pattern configured byte is
    played out

```

```

        implementationSpecific: payload is replaced according
                                to an implementation-specific
                                mechanism.
    "

    DEFVAL { 1 }
    ::= { teaPwTDMCcfgEntry 12 }

teaPwTDMCcfgAvePktLossTimeWindow OBJECT-TYPE
    SYNTAX      Integer32
    UNITS       "millisecond"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The length of time over which the average packet
         loss rate should be computed to detect Excessive packet
         loss rate"
    ::= { teaPwTDMCcfgEntry 13}

teaPwTDMCcfgExcessivePktLossThreshold OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Excessive packet loss rate is detected by computing the
         average packet loss rate over a
         teaPwTDMCcfgAvePktLossTimeWindow
         amount of time and comparing it with this threshold value. "
    ::= { teaPwTDMCcfgEntry 14 }

teaPwTDMCcfgAlarmThreshold OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Alarms are only reported when the defect state persists
         for the length of time specified by this object.

         The object's unit is millisec"
    DEFVAL { 2500 }
    ::= { teaPwTDMCcfgEntry 15 }

teaPwTDMCcfgClearAlarmThreshold OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Alarm MUST be cleared after the corresponding defect is
         undetected for the amount of time specified by this object.
         The object's unit is millisec"
    DEFVAL { 10000 }
    ::= { teaPwTDMCcfgEntry 16 }

teaPwTDMCcfgMissingPktsToSes OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Number of missing packets detected (consecutive or not)
         within a 1-second window to cause a Severely Errored
         Second (SES) to be counted."
    DEFVAL { 3 }
    ::= { teaPwTDMCcfgEntry 17 }

teaPwTDMCcfgTimestampMode OBJECT-TYPE
    SYNTAX      INTEGER
                {

```

```

        notApplicable (1),
        absolute      (2),
        differential   (3),
        primeDifferential (4)
    }

MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Timestamp generation MAY be used in one of the following
    modes:
    1. Absolute mode: the PSN-bound IWF sets timestamps
    using the clock recovered from the incoming TDM attachment
    circuit. As a consequence, the timestamps are closely
    correlated with the sequence numbers. All TDM implementations
    that support usage of the RTP header MUST support this mode.
    2. Differential mode: Both IWFs have access to a common high-
    quality timing source, and this source is used for timestamp
    generation. Support of this mode is OPTIONAL.
    3. Prime Differential mode: similar to Differential where
    timestamps are updated according to known constant that is
    prime relative to the packet rate. Support of this mode is
    OPTIONAL. "

::= { teaPwTDMCfgEntry 18 }

teaPwTDMCfgStorageType OBJECT-TYPE
SYNTAX          StorageType
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "This variable indicates the storage type for this
    row."
::= { teaPwTDMCfgEntry 19 }

teaPwTDMCfgFillerPattern OBJECT-TYPE
SYNTAX          Unsigned32 (0..255)
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Filler byte pattern played out on the TDM interface
    if teaPwTDMCfgPktReplacePolicy is set to fillerPattern."

DEFVAL { 255 }

::= { teaPwTDMCfgEntry 20 }

teaPwTDMCfgLflagPayloadPolicy OBJECT-TYPE
SYNTAX          INTEGER
                {
                    ignore (1),
                    use (2),
                }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "This parameter determines whether the payload received
    in packets with L bit set should be ignored or used.

    ignore: discard payload. AIS pattern is played out
    instead of the payload.

    use: do not replace payload. When this option is
    selected no further alarm processing of
    L-bit is performed."

DEFVAL { 1 }
::= { teaPwTDMCfgEntry 21 }

```

```

teaPwTDMCfGIPTos      OBJECT-TYPE
SYNTAX                Unsigned32
MAX-ACCESS            read-create
STATUS                current
DESCRIPTION
    "This variable indicates the TOS value to be used in the IP
    header."

 ::= { teaPwTDMCfGEntry 22 }

teaPwTDMCfGRtpPT      OBJECT-TYPE
SYNTAX                Unsigned32 (0..127)
MAX-ACCESS            read-create
STATUS                current
DESCRIPTION
    "This object configures the payload type to use in
    the RTP header of the data frames."

 ::= { teaPwTDMCfGEntry 23 }

teaPwTDMCfGPeerRtpPT OBJECT-TYPE
SYNTAX                Unsigned32 (0..127)
MAX-ACCESS            read-create
STATUS                current
DESCRIPTION
    "This object configures the payload type that the peer uses
    in the RTP header of the data frames. When this value is
    non-zero it may be used to check incoming data frames for
    validity. "
DEFVAL { 0 }

 ::= { teaPwTDMCfGEntry 24 }

teaPwTDMCfGRtpTSRef   OBJECT-TYPE
SYNTAX                Unsigned32
MAX-ACCESS            read-create
STATUS                current
DESCRIPTION
    "This object configures the timestamp reference to use.
    This is the number of clocks in a 125- $\mu$ s period.
    For example, the following frequencies are represented
    as follows: 8k(1), 32k(4), 256k(64), 1.544M(193),
    2.048(256), 10.24(1280), 19.44 (2430)"
DEFVAL { 1280 }

 ::= { teaPwTDMCfGEntry 25 }

teaPwTDMCfGPeerRtpTSRef OBJECT-TYPE
SYNTAX                Unsigned32
MAX-ACCESS            read-create
STATUS                current
DESCRIPTION
    "This object configures the timestamp reference
    that the peer is using.
    This is the number of clocks in a 125- $\mu$ s period.
    For example, the following frequencies are represented
    as follows: 8k(1), 32k(4), 256k(64), 1.544M(193),
    2.048(256), 10.24(1280), 19.44 (2430)"
DEFVAL { 1280 }

 ::= { teaPwTDMCfGEntry 26 }

teaPwTDMCfGSRTPenable OBJECT-TYPE
SYNTAX                TruthValue
MAX-ACCESS            read-create
STATUS                current

```

```

DESCRIPTION
    "Object used for enabling use of SRTP for protection
    of payload data"
DEFVAL {false }

::= { teaPwTDMCfGEntry 27 }

-- END of Table

-- PW CESoPSN specific Configuration Table

teaPwTDMCfGFrAmEdInDexNExt OBJECT-TYPE
SYNTAX          TestAndIncr
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "This object contains the value to be used for
    teaPwTDMCfGFrAmEdInDex when creating entries in the
    teaPwTDMCfGFrAmEdTable. The value 0 indicates that no
    unassigned entries are available. To obtain the
    value of teaPwTDMCfGFrAmEdInDexNExt for a new entry in the
    teaPwTDMCfGFrAmEdTable, the manager issues a management
    protocol retrieval operation to obtain the current
    value of teaPwTDMCfGFrAmEdInDexNExt. After each retrieval
    operation, the agent should modify the value to
    reflect the next unassigned index. After a manager
    retrieves a value, the agent will determine through
    its local policy when this index value will be made
    available for reuse."
::= { teaPwTDMObjects 4 }

teaPwTDMCfGFrAmEdTable OBJECT-TYPE
SYNTAX          SEQUENCE OF TeaPwTDMCfGFrAmEdEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table contains a set of parameters that may be
    referenced by one or more CESoPSN TDM PWs in teaPwTDMTable."

::= { teaPwTDMObjects 5 }

teaPwTDMCfGFrAmEdEntry OBJECT-TYPE
SYNTAX          TeaPwTDMCfGFrAmEdEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "These parameters define the characteristics of a
    TDM CESoPSN PW. They are grouped here to ease NMS burden.
    Once an entry is created here, it may be reused
    by many CESoPSN PWs."

INDEX { teaPwTDMCfGFrAmEdInDex }

::= { teaPwTDMCfGFrAmEdTable 1 }

TeaPwTDMCfGFrAmEdEntry ::= SEQUENCE {

    teaPwTDMCfGFrAmEdInDex          TeaPwTDMCfGInDex,
    teaPwTDMCfGFrAmEdRowStatus      RowStatus,
    teaPwTDMCfGFrAmEdIdlePattern    Unsigned32,
    teaPwTDMCfGFrAmEdLflagPolicy    INTEGER,
    teaPwTDMCfGFrAmEdRflagPolicy    INTEGER,
    teaPwTDMCfGFrAmEdRDPolicy      INTEGER,
    teaPwTDMCfGFrAmEdLopsPolicy     INTEGER,
    teaPwTDMCfGFrAmEdSigIPTos      Unsigned32,

```

```

teaPwTDMCfgFramedSigPT                Unsigned32,
teaPwTDMCfgFramedSigPeerPT            Unsigned32,
teaPwTDMCfgFramedSigIdle              Unsigned32,
teaPwTDMCfgFramedSigInterval          Unsigned32,
teaPwTDMCfgFramedSigMaxInterval       Unsigned32
}

```

```

teaPwTDMCfgFramedIndex  OBJECT-TYPE

```

```

SYNTAX          TeaPwTDMCfgIndex
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION

```

```

    "Index to an entry in this table. The value is a copy of the
    assigned teaPwTDMCfgFramedIndexNext"

```

```

 ::= { teaPwTDMCfgFramedEntry 1 }

```

```

teaPwTDMCfgFramedRowStatus  OBJECT-TYPE

```

```

SYNTAX          RowStatus
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION

```

```

    "Object used for creating, modifying, and deleting
    a row from this table.
    The row should not be deleted if the entry is in use"

```

```

 ::= { teaPwTDMCfgFramedEntry 2 }

```

```

teaPwTDMCfgFramedIdlePattern  OBJECT-TYPE

```

```

SYNTAX          Unsigned32 (0..255)
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION

```

```

    "Default Idle Pattern played out on the TDM interface
    on various defects, according to policy, including
    receipt of L bit, R and RD bit and when the pseudo
    wire is administratively disabled."

```

```

DEFVAL { 255 }

```

```

 ::= { teaPwTDMCfgFramedEntry 3 }

```

```

teaPwTDMCfgFramedLflagPolicy  OBJECT-TYPE

```

```

SYNTAX          INTEGER
                {
                idle (1),
                trunkAis(2),
                channelIdle(3)
                }

```

```

MAX-ACCESS      read-create
STATUS          current
DESCRIPTION

```

```

    " This parameter determines the signaling information played
    out on the TDM bound interface upon identification of defect
    state through the receipt of L-flag set

```

```

idle: Idle pattern is played out instead of payload

```

```

trunkAis: AIS (all ones) on the entire trunk (T1/E1) is
played out

```

```

ChannelIdle: Idle pattern is played out instead of payload
and Channel idle indication is set up on the trunk
signaling"

```

```

DEFVAL { 1 }

```

```

 ::= { teaPwTDMCfgFramedEntry 4 }

```

```

teaPwTDMCfGframedRflagPolicy OBJECT-TYPE
SYNTAX INTEGER
{
    none          (1),
    rai           (2),
    channelId     (3)
}

MAX-ACCESS read-create
STATUS current
DESCRIPTION
" This parameter determines the information played out
on the TDM-bound interface upon identification of
remote-end defect state through the receipt of
R-flag set.

none: the payload is not modified, and no error signaling
is sent.

rai: The RAI pattern is generated on the link

channelIdle: The idle byte is played out instead of payload
and channel idle indication is set up in the trunk
signaling."
DEFVAL { 1 }
 ::= { teaPwTDMCfGframedEntry 5 }

```

```

teaPwTDMCfGframedRDPolicy OBJECT-TYPE
SYNTAX INTEGER
{
    none          (1),
    rai           (2),
    channelId     (3)
}

MAX-ACCESS read-create
STATUS current
DESCRIPTION
" This parameter determines the information played out on
the TDM-bound interface upon identification of remote-end
defect state through the receipt of 10 M bit and 0 L bit.

none: the payload is not modified, and no error signaling
is sent.

rai: the RAI pattern is generated on the link

channelIdle: The idle byte is played out instead of payload
and channel idle indication is set up on the
trunk signaling."
DEFVAL { 1 }
 ::= { teaPwTDMCfGframedEntry 6 }

```

```

teaPwTDMCfGframedLopsPolicy OBJECT-TYPE
SYNTAX INTEGER
{
    idle          (1),
    trunkAis     (2),
    channelId     (3)
}

MAX-ACCESS read-create
STATUS current
DESCRIPTION
" This parameter determines the information played
out on the TDM bound interface upon identification
of LOPS state

```

```

Idle: Idle pattern is played out.

trunkAis: AIS (all ones) on the entire trunk (T1/E1)
is played out

channelIdle: Idle pattern is played out instead of
payload and 'Channel idle' indication is set up on
the trunk signaling"

    DEFVAL { 1 }
::= { teaPwTDMCfgFramedEntry 7 }

teaPwTDMCfgFramedSigIPTos OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This variable indicates the TOS value to be used in
    the IP header of CAS signaling packets, if present."
::= { teaPwTDMCfgFramedEntry 8 }

teaPwTDMCfgFramedSigPT OBJECT-TYPE
SYNTAX      Unsigned32 (0..127)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This object configures the payload type to use in the
    RTP header of signaling frames."
::= { teaPwTDMCfgFramedEntry 9 }

teaPwTDMCfgFramedSigPeerPT OBJECT-TYPE
SYNTAX      Unsigned32 (0..127)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This object configures the payload type to use in
    the RTP header of signaling frames.
    When this value is non-zero, it may be used to
    check incoming data frames for validity. "
DEFVAL {0 }
::= { teaPwTDMCfgFramedEntry 10 }

teaPwTDMCfgFramedSigIdle OBJECT-TYPE
SYNTAX      Unsigned32 (0..15)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "Default ABCD CAS signaling Idle pattern. This is
    the pattern sent when a failure is detected on the
    TDM interface, including multi-frame failures. "
DEFVAL {15 }
::= { teaPwTDMCfgFramedEntry 11 }

teaPwTDMCfgFramedSigInterval OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This object configures the application signaling
    interval to use when sending the CAS signaling
    in a separate packet stream.
    The object's unit is ms and the default value is 5 ms. "
DEFVAL {5 }
::= { teaPwTDMCfgFramedEntry 12 }

teaPwTDMCfgFramedSigMaxInterval OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-create

```

```

STATUS          current
DESCRIPTION
    "This object configures the application signaling
    time in which to refresh the peer when no application
    signaling has changed in this interval.
    This is only valid when configured to send the CAS
    signaling in a separate packet stream.
    The object's unit is sec and the default value is 5 seconds. "
    DEFVAL {5 }
    ::= { teaPwTDMCfgFramedEntry 13 }

-- END of Table

-- The following counters work together to integrate
-- errors and the lack of errors on the TDM PW. An error is
-- caused by a missing packet. Missing packets can be a result
-- of: packet loss in the network, (uncorrectable) packet out
-- of sequence, packet length error, jitter buffer overflow,
-- and jitter buffer underflow. The result is declaring whether
-- or not the TDM PW is in Loss of Packet (LOPS) state.

-- TDM PW Performance Current Table.

teaPwTDMPerfCurrentTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF TeaPwTDMPerfCurrentEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The current 15-minute interval counts are in
        this table.

        This table provides per TDM PW performance information."

    ::= { teaPwTDMObjects 6 }

teaPwTDMPerfCurrentEntry OBJECT-TYPE
    SYNTAX          TeaPwTDMPerfCurrentEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created by the agent for every
        teaPwTDM entry. After 15 minutes, the contents of this table
        entry are copied to a new entry in the teaPwTDMPerfInterval
        table and the counts in this entry are reset to zero."

    INDEX          { teaPwIndex }

    ::= { teaPwTDMPerfCurrentTable 1 }

TeaPwTDMPerfCurrentEntry ::= SEQUENCE {

    teaPwTDMPerfCurrentMissingPkts          PerfCurrentCount,
    teaPwTDMPerfCurrentPktsReOrder         PerfCurrentCount,
    teaPwTDMPerfCurrentJtrBfrUnderruns     PerfCurrentCount,
    teaPwTDMPerfCurrentMisOrderDropped     PerfCurrentCount,
    teaPwTDMPerfCurrentMalformedPkt       PerfCurrentCount,

    teaPwTDMPerfCurrentESS                  PerfCurrentCount,
    teaPwTDMPerfCurrentSESS                 PerfCurrentCount,
    teaPwTDMPerfCurrentUASS                 PerfCurrentCount,
    teaPwTDMPerfCurrentFC                   PerfCurrentCount,
    teaPwTDMPerfCurrentJtrBfrMin            Unsigned32,
    teaPwTDMPerfCurrentJtrBfr              Unsigned32,
    teaPwTDMPerfCurrentJtrBfrMax           Unsigned32

}

```

```

teaPwTDMPerfCurrentMissingPkts OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of missing packets (as detected via control word
        sequence number gaps)."
```

::= { teaPwTDMPerfCurrentEntry 1 }

```

teaPwTDMPerfCurrentPktsReOrder OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets detected out of sequence (via control
        word sequence number), but successfully reordered.
        Note: some implementations may not support this feature."
```

::= { teaPwTDMPerfCurrentEntry 2 }

```

teaPwTDMPerfCurrentJtrBfrUnderruns OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of times a packet is needed to be played
        out and the jitter buffer was empty."
```

::= { teaPwTDMPerfCurrentEntry 3 }

```

teaPwTDMPerfCurrentMisOrderDropped OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets detected out of order(via control word
        sequence numbers), and could not be reordered, or could
        not fit in the jitter buffer."
```

::= { teaPwTDMPerfCurrentEntry 4 }

```

teaPwTDMPerfCurrentMalformedPkt OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets detected with unexpected size, or
        bad headers' stack"
```

::= { teaPwTDMPerfCurrentEntry 5 }

```

teaPwTDMPerfCurrentESs OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter associated with the number of Error
        Seconds encountered."
```

::= { teaPwTDMPerfCurrentEntry 6 }

```

teaPwTDMPerfCurrentSESSs OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter associated with the number of
        Severely Errored Seconds encountered."
```

::= { teaPwTDMPerfCurrentEntry 7 }

```

teaPwTDMPerfCurrentUASSs OBJECT-TYPE
    SYNTAX          PerfCurrentCount
    MAX-ACCESS      read-only
    STATUS          current
```

```

DESCRIPTION
    "The counter associated with the number of
    Unavailable Seconds encountered."
 ::= { teaPwTDMPerfCurrentEntry 8 }

teaPwTDMPerfCurrentFC OBJECT-TYPE
    SYNTAX      PerfCurrentCount
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "TDM Failure Counts (FC-TDM). The number of TDM failure
        events. A failure event begins when the LOPS failure
        is declared, and ends when the failure is cleared. A
        failure event that begins in one period and ends in
        another period is counted only in the period in which
        it begins."
 ::= { teaPwTDMPerfCurrentEntry 9 }

teaPwTDMPerfCurrentJtrBfrMin OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The minimum value of the jitter buffer measured in the
        current measurement interval. The difference between the
        minimum value and maximum value reflects the observed
        jitter in the network.
        The units are (microsec)"
 ::= { teaPwTDMPerfCurrentEntry 10 }

teaPwTDMPerfCurrentJtrBfr OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current value of the jitter buffer.
        The units are (microsec)."
```

```

 ::= { teaPwTDMPerfCurrentEntry 11 }

teaPwTDMPerfCurrentJtrBfrMax OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The maximum value of the jitter buffer measured in the
        current measurement interval. The difference between the
        minimum value and maximum value reflects the observed
        jitter in the network.
        The units are (microsec)."
```

```

 ::= { teaPwTDMPerfCurrentEntry 12 }

-- End TDM PW Performance Current Interval Table

-- TDM PW Performance Interval Table.

teaPwTDMPerfIntervalTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TeaPwTDMPerfIntervalEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table provides performance information per TDM PW
        similar to the teaPwTDMPerfCurrentTable above. However,
        these counts represent historical 15-minute intervals.
        Typically, this table will have a maximum of 96 entries
        for a 24-hour period, but is not limited to this. "
```

```

 ::= { teaPwTDMObjects 7 }

```

```

teaPwTDMPerfIntervalEntry OBJECT-TYPE
SYNTAX      TeaPwTDMPerfIntervalEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "An entry in this table is created by the agent for
    every teaPwTDMPerfCurrentEntry that is 15 minutes old.
    The contents of the Current entry are copied to the new
    entry here. The Current entry, then resets its counts
    to zero for the next current 15-minute interval. "

INDEX { teaPwIndex, teaPwTDMPerfIntervalNumber }

 ::= { teaPwTDMPerfIntervalTable 1 }

TeaPwTDMPerfIntervalEntry ::= SEQUENCE {
    teaPwTDMPerfIntervalNumber      Unsigned32,
    teaPwTDMPerfIntervalValidData   TruthValue,
    teaPwTDMPerfIntervalDuration    Integer32,

    teaPwTDMPerfIntervalMissingPkts PerfIntervalCount,
    teaPwTDMPerfIntervalPktsReOrder PerfIntervalCount,
    teaPwTDMPerfIntervalJtrBfrUnderruns PerfIntervalCount,
    teaPwTDMPerfIntervalMisOrderDropped PerfIntervalCount,
    teaPwTDMPerfIntervalMalformedPkt PerfIntervalCount,

    teaPwTDMPerfIntervaleSSs        PerfIntervalCount,
    teaPwTDMPerfIntervaleSESSs      PerfIntervalCount,
    teaPwTDMPerfIntervalUASS        PerfIntervalCount,
    teaPwTDMPerfIntervalFC          PerfIntervalCount,
    teaPwTDMPerfIntervalJtrBfrMin   Unsigned32,
    teaPwTDMPerfIntervalJtrBfrMax   Unsigned32
}

teaPwTDMPerfIntervalNumber OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "A number (normally between 1 and 96 to cover a 24-hour
    period) which identifies the interval for which the set
    of statistics is available. The interval identified by 1
    is the most recently completed 15-minute interval, and
    the interval identified by N is the interval immediately
    preceding the one identified by N-1. The minimum range of
    N is 1 through 4. The default range is 1 through 32. The
    maximum value of N is 1 through 96."
 ::= { teaPwTDMPerfIntervalEntry 1 }

teaPwTDMPerfIntervalValidData OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This variable indicates if the data for this interval
    is valid."
 ::= { teaPwTDMPerfIntervalEntry 2 }

teaPwTDMPerfIntervalDuration OBJECT-TYPE
SYNTAX      Integer32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The duration of a particular interval in seconds,
    Adjustments in the system's time-of-day clock may
    cause the interval to be greater or less than the
    normal value. Therefore this actual interval value
    is provided."
 ::= { teaPwTDMPerfIntervalEntry 3 }

```

```

teaPwTDMPerfIntervalMissingPkts OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of missing packets (as detected via control
        word sequence number gaps)."
```

::= { teaPwTDMPerfIntervalEntry 4 }

```

teaPwTDMPerfIntervalPktsReOrder OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets detected out of sequence (via control
        word sequence number), but successfully reordered.
        Note: some implementations may not support this
        feature."
```

::= { teaPwTDMPerfIntervalEntry 5 }

```

teaPwTDMPerfIntervalJtrBfrUnderruns OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of times a packet needed to be played
        out and the jitter buffer was empty."
```

::= { teaPwTDMPerfIntervalEntry 6 }

```

teaPwTDMPerfIntervalMisOrderDropped OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets detected out of order(via control word
        sequence numbers), and could not be reordered, or could
        not fit in the jitter buffer."
```

::= { teaPwTDMPerfIntervalEntry 7 }

```

teaPwTDMPerfIntervalMalformedPkt OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets detected with unexpected size, or
        bad headers' stack"
```

::= { teaPwTDMPerfIntervalEntry 8 }

```

teaPwTDMPerfIntervalESs OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter associated with the number of Error
        Seconds encountered."
```

::= { teaPwTDMPerfIntervalEntry 9 }

```

teaPwTDMPerfIntervalSEsSs OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The counter associated with the number of
        Severely Errored Seconds encountered."
```

::= { teaPwTDMPerfIntervalEntry 10 }

```

teaPwTDMPerfIntervalUASs OBJECT-TYPE
    SYNTAX          PerfIntervalCount
    MAX-ACCESS      read-only
```

```

STATUS          current
DESCRIPTION
    "The counter associated with the number of
      Unavailable Seconds encountered."
 ::= { teaPwTDMPerfIntervalEntry 11 }

teaPwTDMPerfIntervalFC OBJECT-TYPE
SYNTAX          PerfIntervalCount
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "TDM Failure Counts (FC-TDM). The number of TDM failure
      events. A failure event begins when the LOPS failure
      is declared, and ends when the failure is cleared. A
      failure event that begins in one period and ends in
      another period is counted only in the period in which
      it begins."
 ::= { teaPwTDMPerfIntervalEntry 12 }

teaPwTDMPerfIntervalJtrBfrMin OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The minimum value of the jitter buffer measured in the
      measurement interval.
      The units are (microsec)"
 ::= { teaPwTDMPerfIntervalEntry 13 }

teaPwTDMPerfIntervalJtrBfrMax OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The maximum value of the jitter buffer measured in the
      measurement interval.
      The units are (microsec)."
 ::= { teaPwTDMPerfIntervalEntry 14 }

-- End TDM PW Performance Interval Table

-- TDM PW Performance 1 Day Interval Table.

teaPwTDMPerf1DayIntervalTable OBJECT-TYPE
SYNTAX          SEQUENCE OF TeaPwTDMPerf1DayIntervalEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table provides per-TDM PW Performance information for
      the current day measurement and the previous days."
 ::= { teaPwTDMObjects 8 }

teaPwTDMPerf1DayIntervalEntry OBJECT-TYPE
SYNTAX          TeaPwTDMPerf1DayIntervalEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "An entry in this table is created by the agent for every
      PW."
INDEX          { teaPwIndex, teaPwTDMPerf1DayIntervalNumber }

 ::= { teaPwTDMPerf1DayIntervalTable 1 }

TeaPwTDMPerf1DayIntervalEntry ::= SEQUENCE {
    teaPwTDMPerf1DayIntervalNumber          Unsigned32,
    teaPwTDMPerf1DayIntervalValidData      TruthValue,
    teaPwTDMPerf1DayIntervalMoniSecs       HCPerfTimeElapsed,
    teaPwTDMPerf1DayIntervalMissingPkts    Counter32,
    teaPwTDMPerf1DayIntervalPktsReOrder    Counter32,

```

teaPwTDMPerf1DayIntervalJtrBfrUnderruns	Counter32,
teaPwTDMPerf1DayIntervalMisOrderDropped	Counter32,
teaPwTDMPerf1DayIntervalMalformedPkt	Counter32,
teaPwTDMPerf1DayIntervalESS	Counter32,
teaPwTDMPerf1DayIntervalSESS	Counter32,
teaPwTDMPerf1DayIntervalUASS	Counter32,
teaPwTDMPerf1DayIntervalFC	Counter32,
teaPwTDMPerf1DayIntervalDiscontinuityTime	TimeStamp,
teaPwTDMPerf1DayIntervalJtrBfrMin	Unsigned32,
teaPwTDMPerf1DayIntervalJtrBfrMax	Unsigned32
}	

teaPwTDMPerf1DayIntervalNumber OBJECT-TYPE

SYNTAX Unsigned32(1..31)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"History Data Interval number. Interval 1 is the current day measurement period, Interval 2 is the most recent previous day; interval 30 is 31 days ago. This table provides performance information per TDM PW similar to the teaPwTDMPerfIntervalTable above. However, these counts represent historical 1 day intervals of up to 1 month. This table will have a maximum of 31 entries. This table contains live data, as such, it is NOT persistent."

::= { teaPwTDMPerf1DayIntervalEntry 1 }

teaPwTDMPerf1DayIntervalValidData OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This variable indicates if the data for this interval is valid."

::= { teaPwTDMPerf1DayIntervalEntry 2 }

teaPwTDMPerf1DayIntervalMoniSecs OBJECT-TYPE

SYNTAX HCPerfTimeElapsed

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The amount of time in the 1-day interval over which the performance monitoring information is actually counted. This value will be the same as the interval duration except in a situation where performance monitoring data could not be collected for any reason or agent clock adjustments."

::= { teaPwTDMPerf1DayIntervalEntry 3 }

teaPwTDMPerf1DayIntervalMissingPkts OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of missing packets (as detected via control word sequence number gaps)."

::= { teaPwTDMPerf1DayIntervalEntry 4 }

teaPwTDMPerf1DayIntervalPktsReOrder OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of packets detected out of sequence (via control word sequence number), but successfully re-ordered. Note: some implementations may not support this feature."

::= { teaPwTDMPerf1DayIntervalEntry 5 }

```

teaPwTDMPerf1DayIntervalJtrBfrUnderruns OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Number of times a packet needed to be played
    out and the jitter buffer was empty."
 ::= { teaPwTDMPerf1DayIntervalEntry 6 }

teaPwTDMPerf1DayIntervalMisOrderDropped OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Number of packets detected out of order(via control word
    sequence numbers), and could not be reordered, or could
    not fit in the jitter buffer."
 ::= { teaPwTDMPerf1DayIntervalEntry 7 }

teaPwTDMPerf1DayIntervalMalformedPkt OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Number of packets detected with unexpected size, or
    bad headers' stack."
 ::= { teaPwTDMPerf1DayIntervalEntry 8 }

teaPwTDMPerf1DayIntervaleSSs OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The counter associated with the number of Error
    Seconds encountered. See [RFC 4553] for
    definition of ES."
 ::= { teaPwTDMPerf1DayIntervalEntry 9 }

teaPwTDMPerf1DayIntervaleSESSs OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The counter associated with the number of Severely
    Error Seconds. Also see [RFC 4553] for definition."
 ::= { teaPwTDMPerf1DayIntervalEntry 10 }

teaPwTDMPerf1DayIntervalUASs OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The counter associated with the number of
    Unavailable Seconds.

    NOTE: When first entering the UAS state, the number
    of SES To UAS is added to this object, then as each
    additional UAS occurs, this object increments by one."
 ::= { teaPwTDMPerf1DayIntervalEntry 11 }

teaPwTDMPerf1DayIntervalFC OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "TDM Failure Counts (FC-TDM). The number of TDM failure
    events. A failure event begins when the LOPS failure

```

```

        is declared, and ends when the failure is cleared."
 ::= { teaPwTDMPerf1DayIntervalEntry 12 }

teaPwTDMPerf1DayIntervalDiscontinuityTime OBJECT-TYPE
SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current

DESCRIPTION
    "The value of sysUpTime on the most recent occasion at
    which any one or more of this segment's Counter32
    suffered a discontinuity. If no such
    discontinuities have occurred since the last re-
    initialization of the local management subsystem, then
    this object contains a zero value."
 ::= { teaPwTDMPerf1DayIntervalEntry 13 }

teaPwTDMPerf1DayIntervalJtrBfrMin    OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The minimum value of the jitter buffer measured in the
    measurement interval.
    The units are (microsec)"
 ::= { teaPwTDMPerf1DayIntervalEntry 14 }

teaPwTDMPerf1DayIntervalJtrBfrMax    OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The maximum value of the jitter buffer measured in the
    measurement interval.
    The units are (microsec)."
```

```

 ::= { teaPwTDMPerf1DayIntervalEntry 15 }

-- Conformance Information

teaPwTDMGroups          OBJECT IDENTIFIER ::= { teaPwTDMConformance 1 }
teaPwTDMCompliances     OBJECT IDENTIFIER ::= { teaPwTDMConformance 2 }

bsodPwTDMModuleCompliance MODULE-COMPLIANCE

    STATUS      current
    DESCRIPTION
        "The compliance statement for an agent that supports the
        Docsis TDM emulation specification as part of the
        Business Services over Docsis initiative."

    MODULE -- this module
        MANDATORY-GROUPS { teaPwTDMGroup,
                            teaPwTDMPerfCurrentGroup,
                            teaPwTDMPerfIntervalGroup,
                            teaPwTDMCfgrFramedGroup
                            }

        OBJECT teaPwTDMRate
            MIN-ACCESS read-only
            DESCRIPTION
                "The ability to modify this object is
                not required"
```

OBJECT teaPwTDMIfIndex  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwGenTDMCfgIndex  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwRelTDMCfgIndex  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMConfigError  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgPayloadSize  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgRtpHdrUsed  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to control the RTP header usage is not required."

OBJECT teaPwTDMCfgPktReorder  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to set the packet reordering is not required."

OBJECT teaPwTDMCfgJtrBfrDepth  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgPayloadSuppression  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to set this object is not required."

OBJECT teaPwTDMCfgConsecPktsInSynch  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgConsecMissPktsOutSynch  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgSetUp2SynchTimeOut  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgPktReplacePolicy  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgAvePktLossTimeWindow  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgExcessivePktLossThreshold  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgAlarmThreshold  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgClearAlarmThreshold  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgMissingPktsToSes  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgTimestampMode  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgStorageType  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfgFillerPattern  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGLflagPayloadPolicy  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGIPTos  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGRtpPT  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGPeerRtpPT  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGRtpTSRef  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGPeerRtpTSRef  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGSRTPenable  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGFramedIdlePattern  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGFramedLflagPolicy  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGFramedRflagPolicy  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGFramedRDPolicy  
 MIN-ACCESS read-only  
 DESCRIPTION  
 "The ability to modify this object is not required"

OBJECT teaPwTDMCfGFramedLopsPolicy  
 MIN-ACCESS read-only

```

        DESCRIPTION
            "The ability to modify this object
            is not required"

OBJECT teaPwTDMCfgFramedSigIPTos
MIN-ACCESS read-only
DESCRIPTION
    "The ability to modify this object
    is not required"

OBJECT teaPwTDMCfgFramedSigPT
MIN-ACCESS read-only
DESCRIPTION
    "The ability to modify this object is not
    required"

OBJECT teaPwTDMCfgFramedSigPeerPT
MIN-ACCESS read-only
DESCRIPTION
    "The ability to modify this object is not
    required"

OBJECT teaPwTDMCfgFramedSigIdle
MIN-ACCESS read-only
DESCRIPTION
    "The ability to modify this object
    is not required"

OBJECT teaPwTDMCfgFramedSigInterval
MIN-ACCESS read-only
DESCRIPTION
    "The ability to modify this object
    is not required"

OBJECT teaPwTDMCfgFramedSigMaxInterval
MIN-ACCESS read-only
DESCRIPTION
    "The ability to modify this object
    is not required"

 ::= { teaPwTDMCompliances 1 }

-- Units of conformance.

teaPwTDMGroup OBJECT-GROUP
OBJECTS {
    teaPwTDMRate,
    teaPwTDMIfIndex,
    teaPwGenTDMCfgIndex,
    teaPwRelTDMCfgIndex,
    teaPwTDMConfigError,
    teaPwTDMTimeElapsed,
    teaPwTDMValidIntervals,
    teaPwTDMCurrentIndications,
    teaPwTDM LatchedIndications,
    teaPwTDMLastEsTimeStamp,
    teaPwTDMRtpSSRC,

    teaPwTDMCfgIndexNext,

    teaPwTDMCfgRowStatus,
    teaPwTDMCfgConfErr,
    teaPwTDMCfgPayloadSize,
    teaPwTDMCfgPktReorder,
    teaPwTDMCfgRtpHdrUsed,
    teaPwTDMCfgJtrBfrDepth,
    teaPwTDMCfgPayloadSuppression,
    teaPwTDMCfgConsecPktsInSynch,
    teaPwTDMCfgConsecMissPktsOutSynch,
    teaPwTDMCfgSetUp2SynchTimeOut,

```

```

        teaPwTDMCfgPktReplacePolicy,

        teaPwTDMCfgAvePktLossTimeWindow,
        teaPwTDMCfgExcessivePktLossThreshold,

        teaPwTDMCfgAlarmThreshold,
        teaPwTDMCfgClearAlarmThreshold,
        teaPwTDMCfgMissingPktsToSes,
        teaPwTDMCfgTimestampMode,
        teaPwTDMCfgStorageType,

        teaPwTDMCfgFillerPattern,
        teaPwTDMCfgLflagPayloadPolicy,
        teaPwTDMCfgIPTos,
        teaPwTDMCfgRtpPT,
        teaPwTDMCfgRtpTSRef
    }
    STATUS current
    DESCRIPTION
        "Collection of objects for basic TDM PW config and
        status."
    ::= { teaPwTDMGroups 1 }

teaPwTDMPerfCurrentGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMPerfCurrentMissingPkts,
        teaPwTDMPerfCurrentPktsReOrder,
        teaPwTDMPerfCurrentJtrBfrUnderruns,
        teaPwTDMPerfCurrentMisOrderDropped,
        teaPwTDMPerfCurrentMalformedPkt,

        teaPwTDMPerfCurrentESS,
        teaPwTDMPerfCurrentSESS,
        teaPwTDMPerfCurrentUASs,
        teaPwTDMPerfCurrentFC
    }
    STATUS current
    DESCRIPTION
        "Collection of current statistics objects for TDM PWs."
    ::= { teaPwTDMGroups 2 }

teaPwTDMPerfCurrentJtrGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMPerfCurrentJtrBfrMin,
        teaPwTDMPerfCurrentJtrBfr,
        teaPwTDMPerfCurrentJtrBfrMax
    }
    STATUS current
    DESCRIPTION
        "Collection of current statistics of the jitter buffer for PWs.
        This group is optional."
    ::= { teaPwTDMGroups 3 }

teaPwTDMPerfIntervalGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMPerfIntervalValidData,
        teaPwTDMPerfIntervalDuration,

        teaPwTDMPerfIntervalMissingPkts,
        teaPwTDMPerfIntervalPktsReOrder,
        teaPwTDMPerfIntervalJtrBfrUnderruns,
        teaPwTDMPerfIntervalMisOrderDropped,
        teaPwTDMPerfIntervalMalformedPkt,

        teaPwTDMPerfIntervalESS,
        teaPwTDMPerfIntervalSESS,
        teaPwTDMPerfIntervalUASs,

```

```

        teaPwTDMPerfIntervalFC
    }
    STATUS current
    DESCRIPTION
        "Collection of Interval statistics objects for TDM PWS."
    ::= { teaPwTDMGroups 4 }

teaPwTDMPerfIntervalJtrGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMPerfIntervalJtrBfrMin,
        teaPwTDMPerfIntervalJtrBfrMax
    }
    STATUS current
    DESCRIPTION
        "Collection of Interval statistics objects for
        the jitter buffer on TDM PWS.
        This group is optional"
    ::= { teaPwTDMGroups 5 }

teaPwTDMCfgFramedGroup OBJECT-GROUP
    OBJECTS {

        teaPwTDMCfgFramedRowStatus,
        teaPwTDMCfgFramedIdlePattern,
        teaPwTDMCfgFramedLflagPolicy,
        teaPwTDMCfgFramedRflagPolicy,
        teaPwTDMCfgFramedRDPolicy,
        teaPwTDMCfgFramedLopsPolicy,
        teaPwTDMCfgFramedSigIPTos,
        teaPwTDMCfgFramedSigPT,
        teaPwTDMCfgFramedSigIdle,
        teaPwTDMCfgFramedSigInterval,
        teaPwTDMCfgFramedSigMaxInterval

    }
    STATUS current
    DESCRIPTION
        "TDM CESoPSN specific configuration objects."
    ::= { teaPwTDMGroups 6 }

teaPwTDMCfgPeerGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMCfgPeerRtpTSRef,
        teaPwTDMCfgPeerRtpPT,
        teaPwTDMCfgFramedSigPeerPT,
        teaPwTDMPeerRtpSSRC
    }
    STATUS current
    DESCRIPTION
        "Optional TDM RTP configuration objects of Peer."
    ::= { teaPwTDMGroups 7 }

teaPwRtpTDMCfgSRTPGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMCfgSRTPenable
    }
    STATUS current
    DESCRIPTION
        "Optional SRTP configuration objects."
    ::= { teaPwTDMGroups 8 }

teaPwTDMPerf1DayIntervalGroup OBJECT-GROUP
    OBJECTS {
        teaPwTDMPerf1DayIntervalValidData,
        teaPwTDMPerf1DayIntervalMoniSecs,
        teaPwTDMPerf1DayIntervalMissingPkts,
        teaPwTDMPerf1DayIntervalPktsReOrder,
        teaPwTDMPerf1DayIntervalJtrBfrUnderruns,
        teaPwTDMPerf1DayIntervalMisOrderDropped,

```

```

        teaPwTDMPerf1DayIntervalMalformedPkt,

        teaPwTDMPerf1DayIntervalESs,
        teaPwTDMPerf1DayIntervalSESSs,
        teaPwTDMPerf1DayIntervalUASSs,
        teaPwTDMPerf1DayIntervalFC,
        teaPwTDMPerf1DayIntervalDiscontinuityTime
    }
STATUS current
DESCRIPTION
    "TDM One Day Statistics."
 ::= { teaPwTDMGroups 9 }

teaPwTDMPerf1DayIntervalJtrGroup OBJECT-GROUP
OBJECTS {
    teaPwTDMPerf1DayIntervalJtrBfrMin,
    teaPwTDMPerf1DayIntervalJtrBfrMax
}
STATUS current
DESCRIPTION
    "Optional Jitter Buffer One Day Statistics."
 ::= { teaPwTDMGroups 10 }

END

```

## Appendix I

### Operator recommendations

(This appendix does not form an integral part of this Recommendation)

This appendix pulls together information operators can use to construct and manage systems carrying T1/E1 bearing traffic. This Recommendation outlines performance requirements for error rates, availability, etc., that may be difficult to achieve without attention to many operating details. This appendix is not meant to contain an exhaustive list of the operating details germane to successful operation. Rather this appendix is meant to highlight relevant methods and operating practices that could be overlooked.

#### I.1 Preferential treatment for TE-CMs

[ITU-T J.122] and DSG [b-ITU-T J.128] together describe two TLVs that can be used to improve the performance of T1-bearing systems. The paragraphs in the subclauses just below are taken from [b-ITU-T J.128] and slightly modified. See the two referenced Recommendations for technical details.

##### I.1.1 Segregation of T1 traffic

Traditional Telco T1 signals must function with *very low error rates*. To further pave the way for the transport of T1 signals over DOCSIS, it is desirable to be able to segregate upstream T1 signals on a specifically "chosen upstream" (excluding all other CMs). The chosen US may, for instance, have less noise on it. [ITU-T J.122] defines TLV 19 (for UCD messages) which regulates the usage of an US channel. Operators may use this new capability in the following way.

Operators can arrange it so that non-T1 bearing CMs (which do monitor TLV 19) can be held off the chosen US. CMs that do not monitor TLV 19 and choose to register on the chosen upstream may be issued an "Abort Ranging" message. Non-DSG CMs will move to try another US. DSG CMs will drop into one-way mode until Tdsg3 times out after 5 minutes (default) before trying again.

##### I.1.2 Preferential ranging opportunities

Further, traditional Telco T1 signals must function with *very high availability*. It is desirable to allow T1-bearing CMs to range before other CMs in the event of power outages.

[ITU-T J.122] defines a mechanism whereby classes of devices can be preferentially delayed in initial ranging. Cable devices will have a default Ranging Class ID, based on the device type, which could be used to cause them to inhibit initial ranging after an outage. A bit-field TLV (18) is added to the UCD message to indicate which device classes are to inhibit ranging. Operators can help T1-bearing CMs to range first by delaying other CMs.

#### I.2 Service level agreement considerations

Providers of T1 services often maintain service level agreements (SLAs) with subscribers. These agreements are fairly standard but will be determined by individual operators as part of their offering. This clause outlines the typical requirements as compiled from 3-4 existing SLAs from non-operator T1 service suppliers. The general details mentioned herein are only meant to serve as a guideline for the control of a T1-bearing system, with an eye towards minimizing impact to revenue. Said another way, the following list of performance parameters may be directly important to revenue and merit particular attention. Some parameters are operational, and some are more technical. In particular, operators should strive to maximize service availability and minimize maximum latency (in each calendar month).

- Operational considerations:
  - Delivery time for the installation of service – in days After Receipt of Order.
  - Time to restore services in the event of outages – in hours.
- Technical considerations:
  - Maximum latency roundtrip (in any one month) – in milliseconds.
  - Availability (in any one month) – in percent.

The measurement of this parameter is out of scope.

It is a common practice to schedule routine service intervals in advances. The SLA often provides for such coordination and excuses the supplier for such outages. Further, the time used in such scheduled service intervals is often subtracted so the statistical parameters (used to judge SLA performance) are not affected.

### **I.3 Standard DOCSIS operations**

Some standard DOCSIS operations might adversely affect T1 availability and error rates. While CMTS systems have traditionally supported service flows that may be interrupted, voice and T1 services cannot be interrupted casually. Any change to the modem or services provided to subscribers must be done without impeding the ability of the two TE-CMs to forward packets over their link. Further attention may need to be paid to the manner in which the following operations are carried out.

#### **I.3.1 Downloading**

Many different downloading mechanisms have been specified for CMs. Unless a TE-CM can maintain continuous, error-free T1 service throughout a downloading cycle, then a mechanism should be found to perform the operation at a time that minimizes disruption to the service.

#### **I.3.2 Scheduled service**

As mentioned at the end of the last clause, scheduled maintenance on the underlying DOCSIS system should be coordinated with subscribers to T1-bearing services.

#### **I.3.3 Provisioning**

To maximize the availability of the service, initialization delays should be minimized so the service recovers quickly from power outages.

##### **I.3.3.1 IP addresses**

Delays should be minimized as much as possible while obtaining IP addresses. Clause 6.7 covers this in more detail.

##### **I.3.3.2 Delays**

Try to minimize other delays that may occur during initialization during:

- Ranging.
- Establishing time of day.
- Transfer of operational parameters.
- Registration.
- Baseline privacy initialization.

### **I.3.4 Balancing operations**

Operations designed to balance the network should be avoided if they would disrupt service. Such operations might best be performed during scheduled maintenance intervals per the SLA (see clause I.2). If such operations are performed, some attention should be paid to using the quickest method available.

#### **I.3.4.1 Load balancing**

Load balancing operations affecting TE-CMs should be avoided. The provisioning of T1 services may require the static assignment of downstream and upstream to a TE-CM. This may be done, for instance, to decrease the bit-error rates of the TE-CM data communications (see clause I.1.1).

#### **I.3.4.2 Node splitting and maintenance**

In a similar manner, plant maintenance, like node splitting, should not be carried out without considering the effect on T1-bearing services.

## Appendix II

### Delay aspects

(This appendix does not form an integral part of this Recommendation)

As commercial services are frequently used by delay sensitive applications such as voice, it is critical that its delay through the network be kept within the acceptable bounds of such applications. While such delay constraint comes often as an end-to-end requirement, it may be translated into delay constraints for each network segments along the traffic path, to facilitate the network planning and management.

In this appendix, delays experienced by TDM emulation traffic through a TDM-E system are described. For delay sensitive applications such as cell site backhaul, it is desirable to keep the one-way delay through the TDM-E system, for example, the one illustrated in Figure 5-5, below 7 ms. Therefore, it is important to understand and minimize the delay within each network element and at their interfaces. Excessive delay may result in degradation of quality of service (e.g., incurring voice echo) and limit the range of applications TDM-E can support.

In cell site backhaul applications, calls may be dropped (as cell phones move between cells) if the delay is too large. At this time, it is believed that there are no definitive specifications governing the maximum allowable delay. Cell site operators may each have their own requirements. Anecdotally, the tightest spec mentioned is that the one-way delay must be kept below 7 milliseconds; others cite 10 milliseconds. Additionally, it may be important to keep the delays as close to equal as possible for all cell sites.

In a M-CMTS or DOCSIS 3.0 network, the DOCSIS timestamp may be common across the network if DTI Servers are deployed with GPS. If this is true, then delay may be measured between any two DOCSIS devices using DLM (DEPI latency measurement) as defined in [ITU-T J.212] or DPV (DOCSIS path verify) as defined in [ITU-T J.222.2]. This may be useful to understand what segments of the network are contributing delays.

#### II.1 Upstream delay

##### II.1.1 Packetization delay

Each TDM circuit is sampled periodically and the bits form the payload of a packet. The packetization delay is the time between the arrival of the first bit and that of the last bit in the same packet from the TDM circuit. Thus this delay is the same as the sample interval.

##### II.1.2 Processing delay in TE-CM

The eTEA inside the TE-CM packages the payload bits into a packet for transmission to the CMTS. The delay inside TE-CM includes the UGS scheduling delay (see [ITU-T J.122]), the interleaver delay and the transmission delay.

###### II.1.2.1 UGS scheduling delay

The UGS grants for TDM emulation traffic are periodical. In the best case, if the packet is ready right before the UGS grant time there is no additional delay. In the worst case, if the packet just misses the UGS grant time there is an additional delay of one UGS grant interval. The TE-CM should minimize this delay.

###### II.1.2.2 TDMA byte interleaver delay

For TDMA, interleaver may be used in the upstream direction to reduce the packet error rate. The Interleaver delay depends on the interleaver depth and packet size. This delay is not likely to be significant as it only involves packet processing internal to the TE-CM.

### **II.1.2.3 Transmission delay**

The transmission delay onto the wire is:

raw packet size/upstream bandwidth,

where the raw packet size includes all the protocol and physical layer overhead.

### **II.1.3 Propagation delay**

The propagation delay is up to 0.8 ms between the most distant CM and CMTS (see [ITU-T J.122]) but typically much less.

### **II.1.4 Delay in CMTS**

All the TDM emulation traffic should be of high priority. Still a TDM emulation flow may have to contend with other high priority traffic such as high priority control traffic and other TDM emulation flows. Hence a TDM emulation packet may have to be queued behind these types of packets and possibly also a low priority traffic already in transmission. The packet delay will also depend on the queuing architecture and the scheduling algorithm.

The TEA, if residing in a CMTS, may also have a play-back buffer with a depth of more than one packet to absorb the jitter. In such cases, additional buffering delay occurs. See clauses 5.5.2.2.2 and 5.5.2.2.3.

### **II.1.5 M-CMTS delay**

If the upstream RF receiver is physically separated from the core M-CMTS, there will be additional delay due to packet serialization, queuing, transmission, etc.

## **II.2 Downstream delay**

### **II.2.1 Packetization delay**

Similar to the upstream direction, each TDM circuit is sampled periodically and the bits form the payload of a packet. The packetization delay is the time between the arrival of the first bit and that of the last bit in the same packet from the TDM circuit. Thus this delay is the same as the sample interval.

### **II.2.2 Delay in CMTS**

The delay in CMTS includes the queuing delay, the downstream interleaver delay and the transmission delay.

#### **II.2.2.1 Queuing delay**

Although similar in essence to the upstream queuing delay as discussed in clause II.1, the queuing delay in the downstream direction may differ from that of the upstream due to different queuing architecture, scheduling algorithm and contending traffic.

#### **II.2.2.2 Interleaver delay**

The downstream interleaver delay is specified in Table 6-15 of [ITU-T J.122].

#### **II.2.2.3 Transmission delay**

The transmission latency onto the wire is raw packet size/downstream bandwidth where the raw packet size includes all the protocol overhead. The downstream bandwidth is specified in Table B.3 of [ITU-T J.83 and [ITU-T J.210].

### **II.2.3 Propagation delay**

Same as the upstream direction, the propagation delay from RF output to the most distant CM is up to 0.8 ms but typically much less.

## **II.2.4 Delay in TE-CM**

The IWF function, if residing in CMTS, may also have a play-back buffer with a depth of more than one packet to absorb the jitter. In such cases, additional buffering delay occurs.

## **II.2.5 M-CMTS delay**

In the M-CMTS architecture, the core M-CMTS is physically separated from the EQAM and the interface is defined in [ITU-T J.212]. Hence, additional delays exist compared to that in the traditional CMTS architecture. More discussions can be found in Appendix I of [ITU-T J.212].

### **II.2.5.1 Queuing delay in core M-CMTS**

This is similar to the queuing delay in traditional CMTS as described in clause II.2.2.

### **II.2.5.2 PSP packet serialization delay**

PSP mode is recommended for TDM-E because its traffic may be prioritized over other traffic.

If the TDM-E packets are sent to DEPI with PSP mode, there is possible packet serialization delay depending on the packet size. PSP should not serialize TDM-E traffic to avoid incurring this delay.

### **II.2.5.3 Transmission delay onto DEPI**

The transmission latency onto the wire is raw packet size/DEPI bandwidth where the raw packet size includes all the protocol overhead as described in [ITU-T J.212].

### **II.2.5.4 CIN delay**

Delay added by CIN may be noticeable and is under the control of the network operator.

### **II.2.5.5 Latency in EQAM**

Additional queuing delay is possible due to traffic scheduling as illustrated in Figure 6-1 of [ITU-T J.212].

## Appendix III

### Bandwidth usage

(This appendix does not form an integral part of this Recommendation)

In choosing the appropriate packet payload size, one usually has to make tradeoffs between delay and bandwidth utilization. The packet size related delays have been discussed in Appendix II. The bandwidth utilization is discussed here for both upstream and downstream directions.

As described in clause 6.2, the fixed protocol overhead for a TDMoIP packet includes: TDMoIP Control Word (4 bytes), optional RTP header (12 bytes), UDP header (8 bytes), IPv4 header (20 bytes) or IPv6 header (40 bytes). So the fixed per packet overhead added by IWF is 44 bytes for IPv4 and 64 bytes for IPv6.

#### III.1 Upstream bandwidth usage

The raw upstream bandwidth over the DOCSIS interface is a function of the channel width (as given in Table 6-6 of [ITU-T J.122]) and the modulation mode as shown in Table III.1 below.

**Table III.1 – Upstream bandwidth computed from DOCSIS 2.0**

Channel width (MHz)	QPSK (Mbit/s)	8QAM (Mbit/s)	16QAM (Mbit/s)	32QAM (Mbit/s)	64QAM (Mbit/s)
0.2	0.32	0.48	0.64	0.8	0.96
0.4	0.64	0.96	1.28	1.6	1.92
0.8	1.28	1.92	2.56	3.2	3.84
1.6	2.56	3.84	5.12	6.4	7.68
3.2	5.12	7.68	10.24	12.8	15.36
6.4	10.24	15.36	20.48	25.6	30.72

In addition to the fixed per-packet overhead added by the eTEA, DOCSIS in the upstream direction adds an Ethernet header (18 bytes) and a MAC header (6 bytes), which brings the overhead per packet overhead to  $P\_hdr = 68$  bytes for IPv4 and  $P\_hdr = 88$  bytes for IPv6. This overhead may be significantly reduced by deploying PHS but the saving depends on the implementation.

For the structure-agnostic transport, if the packetization interval is  $T_p$  ms, the TDM payload for a T1 circuit is  $P\_tdm = 1.544 \text{ Mbit/s} * T_p \text{ ms} / 8 = 193 * T_p$  bytes, for a E1 circuit is  $P\_tdm = 2.048 \text{ Mbit/s} * T_p \text{ ms} / 8 = 256 * T_p$  bytes.

The physical layer overhead depends on the configured FEC K byte size, FEC T byte size, preamble length  $P\_preamble$  and guard time byte  $P\_guard$  when using TDMA. For the physical layer payload size of  $T\_hdr + T\_tdm$ , the number of whole FEC codeword is  $CW = \text{floor}((T\_hdr + T\_tdm)/K)$ , the number of bytes in the shortened codeword is  $(T\_hdr + T\_tdm) - CW * K$ . The number of shortened codeword is  $CW\_short = \text{ceiling}((T\_hdr + T\_tdm)/K) - CW$  and the total number of bytes is  $CW * (K + 2 * T) + ((T\_hdr + T\_tdm) - CW * K) + 2 * T * CW\_short + P\_preamble + P\_guard$ .

As an example, with  $T = 8$ ,  $K = 20$ ,  $P\_preamble = 20$  and  $P\_guard = 5$ , the raw byte per packet and the raw bandwidth for a UDP/IPv4/RTP encapsulated flow is shown in Table III.2, assuming the use of a shortened codeword. Along with Table III.1, the result of such calculation can be used by the operator for bandwidth provisioning.

**Table III.2 – An Example of upstream bandwidth usage**

Packetization interval (ms)	T1		E1	
	Raw byte per packet	Bandwidth usage (Mbit/s)	Raw byte per packet	Bandwidth usage (Mbit/s)
1	510	4.08	621	4.97
2	847	3.39	1069	4.28
4	1537	3.07	1997	3.99

Based on the upstream cable plant conditions, a network operator may choose the appropriate modulation mode, the PHY layer parameters, and then decide on the packetization interval based on the calculations illustrated above along with the delay budget.

### III.2 Downstream bandwidth usage

#### III.2.1 Downstream RF bandwidth usage

The raw downstream RF bandwidth in systems compliant with [ITU-T J.83] Annex B is 26.97035 Mbit/s for 64QAM and 38.81070 Mbit/s for 256QAM, as shown in Table B.3 of [ITU-T J.83]. The raw downstream RF bandwidth in systems compliant with [ITU-T J.83] Annex A is 41.712 Mbit/s for 64QAM and 55.616 Mbit/s for 256QAM, see [ITU-T J.210].

The downstream PDU and MAC format is the same as the upstream so the overhead is still P\_hdr = 68 bytes for IPv4 and P\_hdr = 88 bytes for IPv6. The DOCSIS payload is then encapsulated in fixed size MPEG frames (183 bytes).

**Table III.3 – An example of downstream bandwidth usage**

Packetization interval (ms)	T1		E1	
	MPEG packet size (Byte)	Bandwidth usage (Mbit/s)	MPEG packet size (Byte)	Bandwidth usage (Mbit/s)
1	376	3.008	376	3.008
2	564	2.256	752	3.008
4	940	1.880	1128	2.256

#### III.2.2 DEPI bandwidth usage

For M-CMTS, the DEPI bandwidth usage can be calculated based on the encapsulation described in [ITU-T J.212] and how the implementation carries the payload using the recommended PSP mode.

## Appendix IV

### eTEA configuration example

#### IV.1 Structure agnostic T1 configuration

Table IV.1 is an example of an eTEA configuration file used to configure a structure agnostic T1 PW.

**Table IV.1 – Structure agnostic T1 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
9	43				Pseudo wire configuration
		1	2	1	PW index 1
		2	1	18	Structure agnostic TDM over IP T1 PW
		3	4	1.2.3.4	Peer's IPv4 address
		5	2	50000	Destination UDP port number
		6	2	50001	Peer Destination UDP port number
		7	4	T1-1	PW name
		9	2	2	PW administratively down
		10	1	0	Notifications not generated when the PW status changes
		12	2	1	This PW uses TDM configuration table 1
		16	3		
		16.1	1	2	This PW is associated with port 2
10	39				Common parameters for TDM configurations
		1	2	1	TDM configuration table 1
		2	2	192	Payload size is 192 bytes
		3	1	0	Optional RTP header is not used
		5	2	5	The jitter buffer is configured to be 5 ms
		6	1	2	Disable payload suppression
		7	2	10	Set the threshold to be 10 for exiting the loss of packet synchronization state
		8	2	4	Set the threshold to be 4 for entering the loss of packet synchronization state
		11	2	5	Window for calculating packet loss rate is set to 5 ms
		12	1	5	Excessive loss occurs if the loss exceeds 5%
		15	1	1	Severe loss occurs if the loss exceeds 1%
		17	1	0xFF	Fill pattern is defined as 0xFF
6	16	Digest value			MD5 Digest
255					End of data marker

## IV.2 Structure agnostic E1 configuration

Table IV.2 is an example of an eTEA configuration file used to configure a structure agnostic E1 PW, with IPv6 addresses for the PW endpoints.

**Table IV.2 – Structure agnostic E1 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
9	55				Pseudo wire configuration
		1	2	1	PW index 1
		2	1	17	Structure agonistic TDM over IP E1 PW
		4	16	2001::1.2.3.4	Peer's IPv6 address
		5	2	50000	Destination UDP port number
		6	2	50001	Peer Destination UDP port number
		7	4	E1-1	PW name
		9	2	2	PW administratively down
		10	1	0	Notifications not generated when the PW status changes
		12	2	1	This PW uses TDM configuration table 1
		16	3		
		16.1	1	2	This PW is associated with port 2
10	35				Common parameters for TDM configurations
		1	2	1	TDM configuration table 1
		2	2	256	Payload size is 256 bytes
		3	1	0	Optional RTP header is not used
		5	2	5	The jitter buffer is configured to be 5 ms
		6	1	2	Disable payload suppression
		7	2	10	Set the threshold to be 10 for exiting the loss of packet synchronization state
		8	2	5	Window for calculating packet loss rate is set to 5 ms
		12	1	5	Excessive loss occurs if the loss exceeds 5%
		15	1	1	Severe loss occurs if the loss exceeds 1%
		17	1	0xFF	Fill pattern is defined as 0xFF
6	16	Digest value			MD5 Digest
255					End of data marker

### IV.3 Structure aware 5xDS0 configuration

Table IV.3 is an example of an eTEA configuration file used to configure a structure aware 5xDS0.

**Table IV.3 – Structure aware nxDS0 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
9	53				Pseudo wire configuration
		1	2	1	PW index 1
		2	1	21	Structure aware PW (CESoPSN)
		3	4	1.2.3.4	Peer's IPv4 address
		5	2	50000	Destination UDP port number
		6	2	50001	Peer Destination UDP port number
		7	4	5DS0	PW name
		9	2	2	PW administratively down
		10	1	0	Notifications not generated when the PW status changes
		12	2	1	This PW uses TDM configuration table 1
		13	2	1	This PW uses CESoPSN configuration table 1
		16	9		
		16.1	1	2	This PW is associated with port 2
		16.2	4	0x1F	The first five timeslots are used
10	39				Common parameters for TDM configurations
		1	2	1	TDM configuration table 1
		2	2	40	Payload size is 40 bytes
		3	1	0	Optional RTP header is not used
		5	2	5	The jitter buffer is configured to be 5 ms
		6	1	2	Disable payload suppression
		7	2	10	Set the threshold to be 10 for exiting the loss of packet synchronization state
		8	2	4	Set the threshold to be 4 for entering the loss of packet synchronization state
		11	2	5	Window for calculating packet loss rate is set to 5 ms
		12	1	5	Excessive loss occurs if the loss exceeds 5%
		15	1	1	Severe loss occurs if the loss exceeds 1%
		17	1	0xFF	Fill pattern is defined as 0xFF
13	16				CESoPSN parameters for TDM configurations
		1	2	1	TDM CESoPSN configuration table 1
		2	1	0xFF	Idle pattern is defined as 0xFF

**Table IV.3 – Structure aware nxDS0 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
		3	1	1	Alarm indication signal on entire trunk is sent if packets are received with L flag set
		4	1	1	When packets are received with the R flag set, remote alarm indication is set
		5	1	0	Do nothing if received packet has L = 0 and M = 01
6	16	Digest value			MD5 Digest
255					End of data marker

#### IV.4 Structure aware 5xDS0 configuration with CAS

Table IV.4 is an example of an eTEA configuration file used to configure a structure aware 5xDS0.

**Table IV.4 – Structure aware nxDS0 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
9	53				Pseudo wire configuration
		1	2	1	PW index 1
		2	1	23	Structure aware PW (CESoPSN) with CAS
		3	4	1.2.3.4	Peer's IPv4 address
		5	2	50000	Destination UDP port number
		6	2	50001	Peer Destination UDP port number
		7	4	5DS0	PW name
		9	2	2	PW administratively down
		10	1	0	Notifications not generated when the PW status changes
		12	2	1	This PW uses TDM configuration table 1
		13	2	1	This PW uses CESoPSN configuration table 1
		16	9		
		16.1	1	2	This PW is associated with port 2
		16.2	4	0x1F	The first five timeslots are used
10	39				Common parameters for TDM configurations
		1	2	1	TDM configuration table 1
		2	2	40	Payload size is 40 bytes
		3	1	0	Optional RTP header is not used
		5	2	5	The jitter buffer is configured to be 5 ms
		6	1	2	Disable payload suppression
		7	2	10	Set the threshold to be 10 for exiting the loss of packet synchronization state
		8	2	4	Set the threshold to be 4 for entering the loss of packet synchronization state

**Table IV.4 – Structure aware nxDS0 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
		11	2	5	Window for calculating packet loss rate is set to 5 ms
		12	1	5	Excessive loss occurs if the loss exceeds 5%
		15	1	1	Severe loss occurs if the loss exceeds 1%
		17	1	0xFF	Fill pattern is defined as 0xFF
13	25				CESoPSN parameters for TDM configurations
		1	2	1	TDM CESoPSN configuration table 1
		2	1	0xFF	Fill pattern is defined as 0xFF
		3	1	1	Idle pattern is sent if packets are received with L flag set
		4	1	1	When packets are received with the R flag set, remote alarm indication is set
		5	1	0	Do nothing if received packet has L = 0 and M = 01
		7	1	5	ToS value for the CAS signalling traffic is 5
		10	1	4	Application signalling idle interval is 4 s
		11	1	4	Application signalling interval is 4 ms
6	16	Digest value			MD5 Digest
255					End of data marker

#### IV.5 Structure aware T1 with RTP configuration

Table IV.5 is an example of an eTEA configuration file used to configure a structure aware T1 with optional RTP.

**Table IV.5 – Structure aware nxDS0 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
9	48				Pseudo wire configuration
		1	2	1	PW index 1
		2	1	21	Structure aware PW (CESoPSN)
		3	4	1.2.3.4	Peer's IPv4 address
		5	2	50000	Destination UDP port number
		6	2	50001	Peer Destination UDP port number
		7	4	5DS0	PW name
		9	2	2	PW administratively down
		10	1	0	Notifications not generated when the PW status changes
		12	2	1	This PW uses TDM configuration table 1
		13	2	1	This PW uses CESoPSN configuration table 1
		18	4	0xABCD	SSRC value

**Table IV.5 – Structure aware nxDS0 PW configuration file example**

Type	Length	Value (subtype)	Length	Value	Description
10	35				Common parameters for TDM configurations
		1	2	1	TDM configuration table 1
		2	2	40	Payload size is 40 bytes
		3	1	1	Optional RTP header is used
		5	2	5	The jitter buffer is configured to be 5 ms
		6	1	2	Disable payload suppression
		7	2	10	Set the threshold to be 10 for exiting the loss of packet synchronization state
		8	2	5	Window for calculating packet loss rate is set to 5 ms
		12	1	5	Excessive loss occurs if the loss exceeds 5%
		15	1	1	Severe loss occurs if the loss exceeds 1%
		17	1	0xFF	Fill pattern is defined as 0xFF
13	16				CESoPSN parameters for TDM configurations
		1	2	1	TDM CESoPSN configuration table 1
		2	1	0xFF	Idle pattern is defined as 0xFF
		3	1	1	Alarm indication signal on entire trunk is sent if packets are received with L flag set
		4	1	1	When packets are received with the R flag set, remote alarm indication is set
		5	1	0	Do nothing if received packet has L = 0 and M = 01
6	16	Digest value			MD5 Digest
255					End of data marker

## Appendix V

### Clocking options, by example

(This appendix does not form an integral part of this Recommendation)

This appendix discusses the various clocking modes supported in the spec and details several examples.

#### V.1 Background

A central element in emulating TDM services is the clocking methodology. This Recommendation allows many different options for clocking the system. This appendix builds on information contained in clauses 5.4 and 5.5 and uses the same terminology. In terms of clocking the TDM links, there are two options: either these links can be tied directly to a Stratum-1 Traceable Reference Source (PRS) or the links provide their own independent timing. When links are tied to a PRS, the TEA at the CMTS uses what is called direct clocking to take the clock from a BITS interface or a DTI interface. The DTI interface can be used in DOCSIS 3.0 and the M-CMTS to provide intra-box synchronization of all of the modes, and this timing reference often includes a PRS reference. On the cable modem side of the network, the DOCSIS 10.24 MHz clock can be recovered in one of two ways: from SYNC messages or from the symbol clock. When TDM links have independent timing, again there are two options. If a reference clock with a common source is available at both ends of the connection, this reference can be used as a reference to timing the link clocks. If no common reference is available, then adaptive clock recovery can be used.

Table V.1 below summarizes legitimate clock modes between the ends of the TDM emulated link; between the ends of the pseudowire. Each column represents a possible configuration between either two TE-CMs or between a TEA associated with a CMTS and a CM.

**Table V.1 – BSoD clocking modes**

Clocking Mode														
Device	NCR		PDCR					ACR						
TE-CMTS	DC		DC					Any	Any	DC	Any	Any	Any	Any
TEA	DC		Adaptive	Loop	Adaptive			DC	Loop			Adaptive		
TE-CM	Through	Through	Adaptive	Adaptive	Loop	Adaptive	Loop	Adaptive	Adaptive	Through	Adaptive	Loop	Loop	
TE-CM(2)	Through		Adaptive				Adaptive		Adaptive					
	tested	tested	tested			tested		tested						

NOTE 1 – When two TE-CMs are used in the configuration, the clock mode of each TDM port is shown.

NOTE 2 – The Through mode implies that the clock is coming from the CU and is the DOCSIS 10.24 MHz clock derived from either the SYNC messages and/or the Symbol clock.

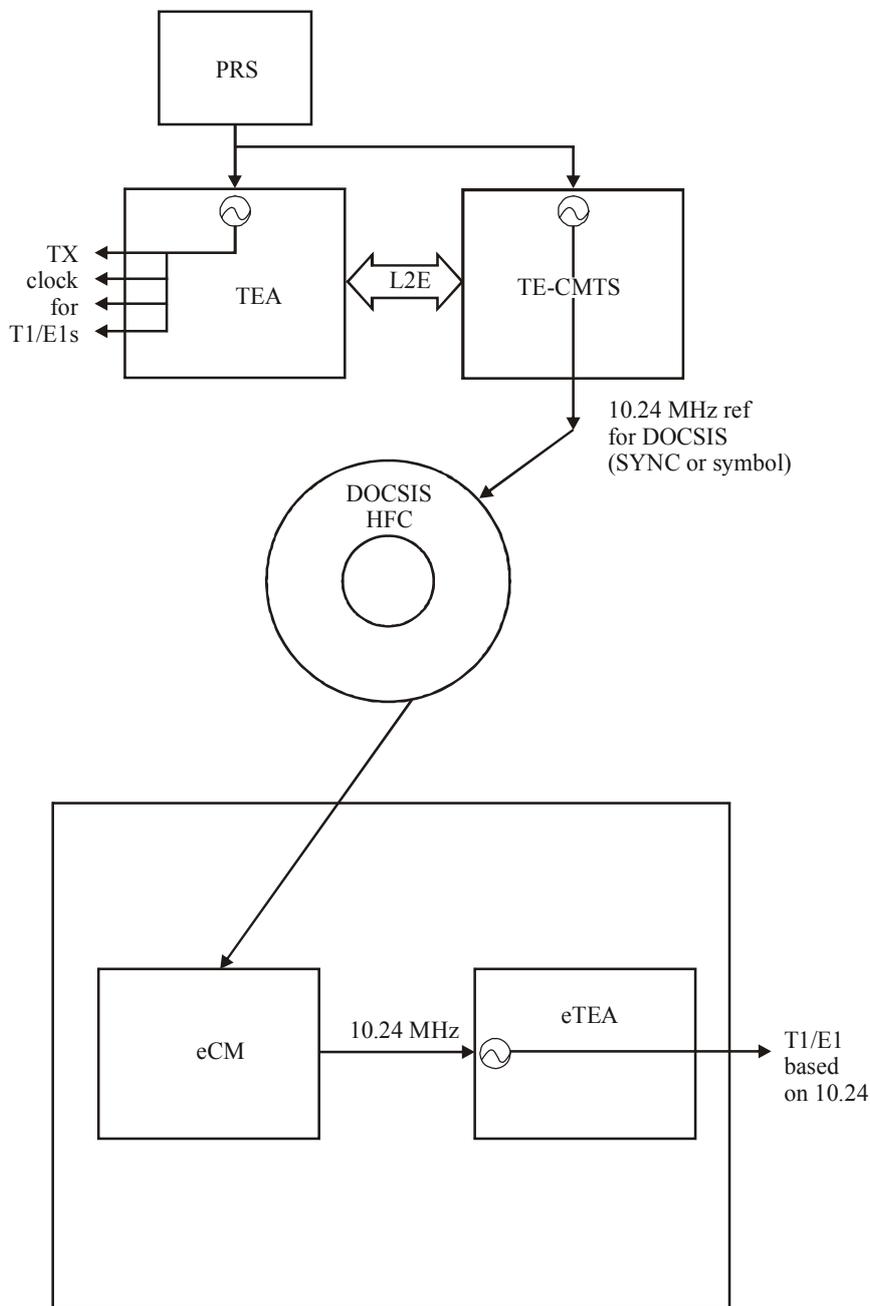
NOTE 3 – Direct clocking (DC) is used in the TE-CMTS and assumes that the clock comes from BITS or from DTI.

NOTE 4 – When PDCR is enabled (together with RTP), this indicates that packets are generated with PDCR timestamps.

## V.2 Clocking modes

### V.2.1 Network clocking

With network clocking, the network clock is used directly as the reference for clocking all of the interfaces. A PLL is required for converting between the frequencies of the references. Figure V.1 below depicts the configuration of the first column of NCR where a PRS clock source is used for clocking all of the TDM interfaces directly. It is recommended that the PRS clock source be a Stratum 1 traceable clock.



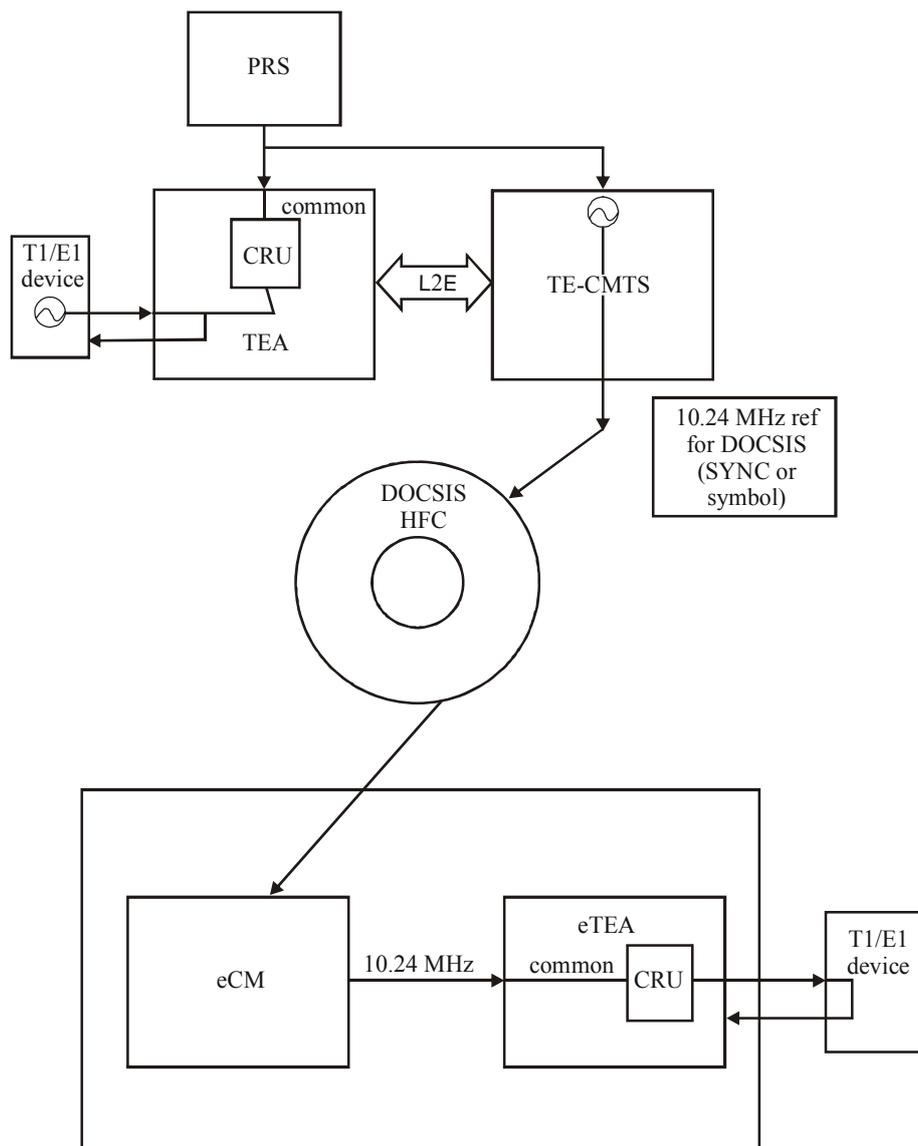
J.214(07)\_F.V-1

Figure V.1 – Network clocking – CMTS to CM

The second column of NCR has a similar picture, except that in place of the T1 being terminated in the TEA; it is terminated in the TE-CM. The clock reference is still taken from the PRS in the core network.

### V.2.2 Prime differential clock recovery

With Prime differential clock recovery, a PRS is used as the reference clock and conveyed to the TE-CMs over the DOCSIS network either using the SYNC messages or using the symbol clock. In this way, both ends of the network are using a common reference clock. Figure V.2 below illustrates the second column of Figure V.1, under PDCR. In this case, the T1 device attached to the TEA provides the timing reference for the T1 service. Information to discern the difference between the two clocks is passed in the RTP, such that the clock can be recovered in the clock recovery unit in the TE-CM.



J.214(07)\_FV-2

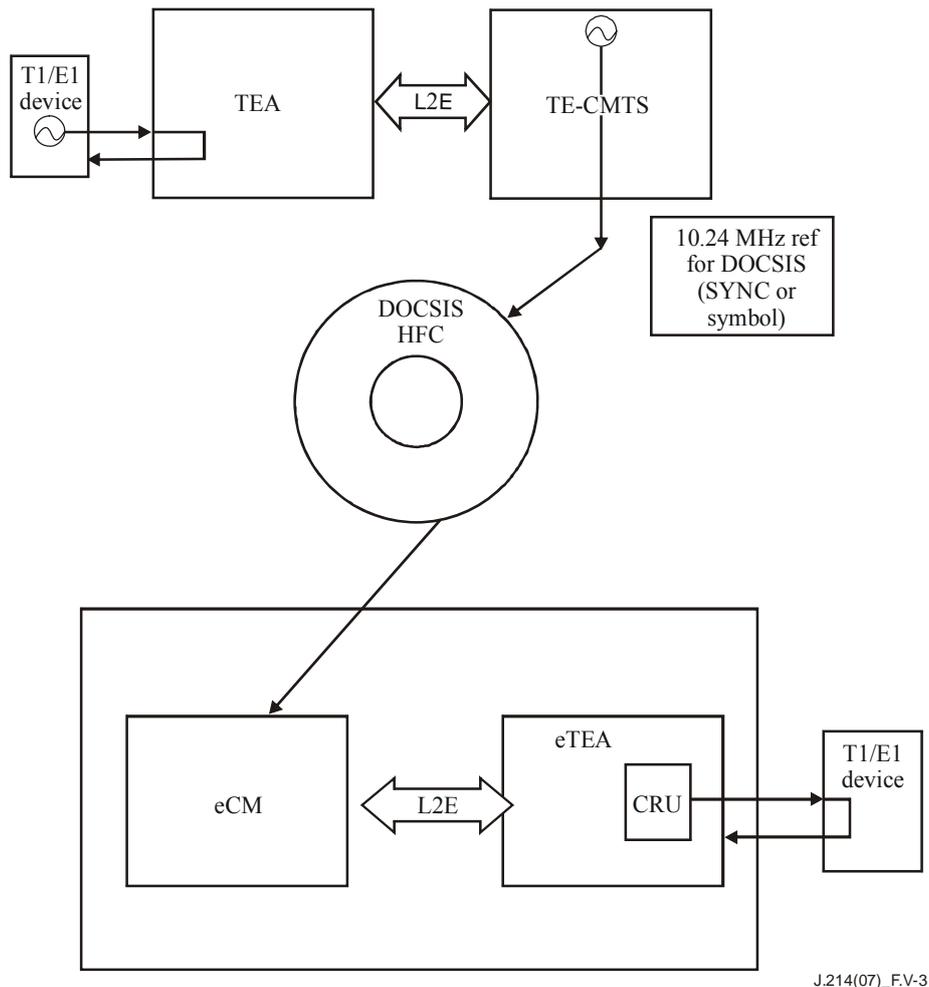
**Figure V.2 – Prime differential clock recovery**

With PDCR, since the operation can be symmetric, other clocking options are supported, as shown in Table V.1. For example, separate clock sources can be supported in each direction, and connection between two TE-CMs can easily be accommodated.

### V.2.3 Adaptive clock recovery

Adaptive clock recovery operates with no common clock between the two ends of the link.

Figure V.3 below depicts the configuration of the second column of Figure V.1 of adaptive clock recovery, where the TEA operates in a loop mode and takes the timing source from the attached T1/E1 device. The CRU within the IWF of the eTEA implements the adaptive clock recovery. The clock is recovered based only on the inter-arrival time of the packets and the recovered clock is sent to the attached E1/T1 device.



**Figure V.3 – Adaptive clock recovery**

Other configuration options of adaptive clock recovery are possible; however, due to the adaptive nature of ACR, care must be taken in setting performance expectations for different configurations. In particular, care must be taken to match the PW packet sizes and rates with the scheduling of the DOCSIS Service flows.

Performance of ACR can be affected by bit error rates and scheduling variations in upstream and downstream data transport. For this reason, planning a service should be done with consideration of the characteristics of both downstream and upstream connections and its relation to the ACR algorithm used.

### V.3 Clocking decision tree

When selecting between the various options, the following simplified decision tree is useful for selecting the best option. These are arranged in a prioritized order:

- 1) Can all nodes in system be synchronized to a single reference source clock (e.g., PRS)?  
Yes – Use network clocking.
- 2) (else) Do the TE-CMTS and TEA support and have access to a common reference clock (e.g., PRS)?  
Yes – Use Prime differential clock recovery.
- 3) (else) Does the TE-CM support adaptive clock recovery?  
Yes – Use adaptive clock recovery. (Preferred mode of operation is to recover clock from TE-CMTS in TE-CM.)

## Bibliography

- [b-ITU-T J.128] ITU-T Recommendation J.128 (2005), *Set-top gateway specification for transmission systems for interactive cable television services*.
- [b-DOCSIS OSSIV3.0] *Data-Over-Cable Service Interface Specifications, DOCSIS 3.0, Operations Support System Interface Specification, CM-SP-OSSIV3.0-I02-070223*, February 23, 2007, Cable Television Laboratories, Inc.
- [b-IETF RFC 5086] IETF RFC 5086 (2007), *Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)*.
- [b-IETF RFC 5087] IETF RFC 5087 (2007), *Time Division Multiplexing over IP (TDMoIP)*.
- [b-MEF 8] Metro Ethernet Forum, *Technical Specification MEF 8, Implementation Agreement for the Emulation of PDH Circuits over Metro Ethernet Networks*, October 2004.



## SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
<b>Series J</b>	<b>Cable networks and transmission of television, sound programme and other multimedia signals</b>
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems