SONET/SDH Circuit Emulation Service Over MPLS (CEM) Management Information Base Using SMIv2
draft-danenberg-pw-cem-mib-00.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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

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

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

1 Abstract

This memo defines an experimental portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects for modeling an adaptation of SONET/SDH circuits over a Multi-Protocol Label Switching (MPLS) [MPLSArch, MPLSFW] Label Switch Router (LSR).
2 Introduction

This document describes a model for managing encapsulated SONET Time Division Multiplexed (TDM) digital signals for transmission over a Packet Switched Network (PSN).

This document is closely related to [CEM], which describes the technology to encapsulate TDM signals and provide the Circuit Emulation Service over a Packet Switched Network (PSN). This document is also related to [TRANS and ENCAP], describing the transport and encapsulation of Layer 2 circuits over MPLS, respectively. NOTE: the CEM encapsulation can be used over PSNs other than MPLS.

The model for CEM management is a MIB. The CEM MIB described in this document works closely with the MIBs described in [PWMIB] and the textual conventions defined in [PWTC]. In the spirit of the [IFMIB], a CEM connection will be a virtual connection (VC), and will therefore not be represented in the ifTable.

Automatic Protection Switching (APS) for CEM is covered within the generic APS modeling within [PWMIB].

CEM is currently specified to carry SONET paths as a "structured" adaptation as well as SONET section/line as "unstructured" adaptation (see Terminology). The SONET section/line interface stack is modeled within [SONETMIB]. The CEM MIB will reference SONET paths (or section/line) as modeled within [SONETMIB].

Note: when transporting SONET section/line in unstructured type, any paths within the section/line are always transported (paths are subordinate to the section/line).

Comments should be made directly to the MPLS mailing list at mpls@uu.net or PWE3 at pwe3@ietf.org.
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [BCP14].

3 Terminology

CEM terminology comes from the CEM draft that describes a mechanism for transporting SONET Time Division Multiplexed (TDM) digital signals over a packet-oriented MPLS network. The mechanism for structured (as outlined in the CEM draft) terminates the SONET section and line overhead and then breaks the SONET path’s Synchronous Payload Envelope (SPE) into fragments for transmission over a PSN. The unstructured mechanism breaks the entire SONET frame (including transport overhead) into fragments for transmission over a PSN. A 32-bit CEM header is appended at the beginning of each fragment to provide information regarding where the SPE begins within the packet stream, a sequence number, and pointer adjustment information.

"Adaptation" refers to the method of adapting a "foreign" communications protocol such that it can be carried by a "native" protocol. In this case, the foreign protocol is SONET/SDH and the native protocol is packet (e.g., MPLS).

"Outbound" references the traffic direction where a SONET path’s payload (SPE) is received, adapted to packet, assigned a VC label, and sent into the PSN.

Conversely, "inbound" is the direction where packets are received from the PSN, packet payloads are reassembled back into an SPE, and inserted as a SONET path into the SONET section and line.

Since a SONET path is bi-directional and symmetrical, it uses the same SONET time-slot, SONET width, packet size. Inbound and outbound VC labels may differ.

CEM will normally transmit into an originating "head" end of a PSN’s tunnel, and receive from a terminating "tail" end a PSN’s tunnel. In the case of an MPLS PSN, a CEM connection typically uses a VC (virtual connection) Label within a Tunnel Label [TRANS]. Multiple CEM VCs each with a unique VC Label and similar traffic engineering requirements can share the same PSN tunnel. For MPLS PSNs, the Tunnel Label is known as the "outer" Label, while the VC Label is known as the "inner" Label. An exception to this is with adjacent MPLS LSRs. In this case, there is an option for CEM VCs to connect directly without an outer Label.

4 The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- An overall architecture, described in RFC 2271 [SNMPArch].
Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIv1 and described in RFC 1155 [SMIv1], RFC 1212 [SNMPv1MIBDef] and RFC 1215 [SNMPv1Traps]. The second version, called SMIv2, is described in RFC 1902 [SMIv2], RFC 1903 [SNMPv2TC] and RFC 1904 [SNMPv2Conf].

Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in RFC 1157 [SNMPv1]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [SNMPv2c] and RFC 1906 [SNMPv2TM]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [SNMPv2TM], RFC 2272 [SNMPv3MP] and RFC 2574 [SNMPv3USM].

Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in RFC 1157 [SNMPv1]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [SNMPv2PO].

A set of fundamental applications described in RFC 2273 [SNMPv3App] and the view-based access control mechanism described in RFC 2575 [SNMPv3VACM].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI. This memo specifies a MIB module that is compliant to the SMIv2. A MIB conforming to the SMIv1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine-readable information in SMIv2 will be converted into textual descriptions in SMIv1 during the translation process. However, this loss of machine-readable information is not considered to change the semantics of the MIB.

4.1 Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object type is named by an OBJECT IDENTIFIER, an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to also refer to the object type.

5 Feature Checklist

This MIB is designed to satisfy the following requirements and constraints:

- Fit within the architecture defined [PWMIB].
- The MIB supports point-to-point CEM connections. Point-to-multipoint connections are for future study.

- The MIB configures the connection: type, packet length, error actions, and many other CEM objects.

- The MIB reports: packet counts, error counts, and many other status objects.

6 CEM MIB usage

6.1 Summary of CEM MIB

- The CEM SONET extension table (pwVcCemSonetExtTable) is used to indicate the time slot of the SONET path to be adapted (if CEM type is structured). This table also contains the index to the pwVc and pwVcCem tables (for reverse mapping). Note: this table is applicable for structured or unstructured CEM types.

- The CEM VC Configuration Parameter Table (pwVcCemConfTable) has entries of CEM VC configuration objects. In situations where config params are common amongst CEM VCs, 1 entry here may be referenced by many mplsCemVcTable entries.

- The CEM VC Table (pwVcCemTable) contains the SONET Path (or Section/Line) ifIndex, the pwVcCemConfTable index, and many objects for monitoring CEM VCs.

- The CEM Performance Current Table (pwVcCemPerfCurrentTable) is an augmentation of pwVcCemTable. It contains CEM stats only for the current 15 minute period.

- The CEM Performance Interval Table (pwVcCemPerfIntervalTable) is similar to the pwVcCemPerfCurrentTable. It contains historical intervals (usually 96 15-minute entries to cover a 24 hour period).

Note: the performance interval statistics are supported by CEM due to the very function of CEM – that is, processing SONET. See [SONETMIB].

6.2 CEM configuration Step by Step

Configuring a CEM VC involves the following steps.

First create an entry in the pwVcTable and configure the PSN tunnels:

- Follow steps as defined in [PWMIB].

Configure the SONET Path parameters (structured only):

- Set the SONET path width in the sonetPathCurrentTable [SONETMIB].
- Set the SONET path starting time slot in the pwVcCemSonetExtTable.

Configure the CEM VC:

- The agent will have created an entry in the pwVcCemTable based on the entry created in the pwVcTable.
- Create an entry in the pwVcCemConfTable set packet length, etc.
- Although MPLS signaling of CEM is outside the scope of this document, LDP parameters are defined in the pwVcCemConfTable: packet length and CEM options. See [TRANS] and [CEM].
- Set the index of this pwVcCemConfTable entry in the pwVcCemTable.
- Once a CEM VC is operational, the pwVcCemTable, pwVcCemPerfCurrentTable, and pwVcCemPerfIntervalTable are used to monitor the various counts, indicators, and conditions of the VC.

7 Example of CEM Setup

In this section we provide an example of using the MIB objects described in section 8 to set up a structured CEM VC (width of 3 STSs, starting at time slot 1). While this example is not meant to illustrate every permutation of the MIB, it is intended as an aid to understanding some of the key concepts. It is meant to be read after going through the MIB itself. See [PWMIB] for an example of setting up Tunnels.

First configure the SONET path width, starting time-slot, and associated CEM VC. In this case, an STS-3c starts at SONET time slot 1 (and is evenly distributed within the SONET frame). In the following example, the ifIndex for both the sonetPathCurrentEntry and mplsCemSonetExtTable is 23, while the pwVcConfTable index is 9.

In sonetPathCurrentEntry (ifIndex = 23) :

```
{  
  sonetPathCurrentWidth           = 3,
  sonetPathCurrentStatus ...  
}
```

In mplsCemSonetExtTable (ifIndex = 23) :

```
{ 
  mplsCemSonetCurrentPathPktLength = 500 -- payload bytes 
}
```

Create an entry in the pwVcCemConfTable (index = 9) :

```
{  
  pwVcCemConfMaxPktLength          = 500 -- payload bytes 
  pwVcCemConfMinPktLength          = 0 -- no minimum 
  pwVcCemConfCemOptions            = 0 -- for LDP signaling 
  pwVcCemConfPktResequence         = false 
  pwVcCemConfEnableECC             = true 
  pwVcCemConfEnableDBA             = allOnesOnAis 
  pwVcCemConfJtrBfrDepth           = 50 -- packets 
}
```
pwVcCemConfConsecPktsInsync = 2  -- Exit LOPS state
pwVcCemConfConsecMissingOutSync = 10 -- Enter LOPS state

Check that there are no error bits set in pwVcCemConfigError.

In PW-MIB: Get a new index and create a new pwVcTable entry using
pwVcIndexNext (here, the VC index = 83) and pwVcRowStatus. In this
ew entry, set pwVcType to 'cem'. This should create a new entry in
the pwVcCemTable. Set up CEM type and indexes within this new
pwVcCem table entry:
{
  pwVcCemType                 = structured
  pwVcCemSonetIfIndex     = 23 -- Index of associated entry
                    -- in sonetPathCurrent table.
  pwVcCemConfIndex          = 9  -- Index of associated entry
                    -- in pwVcCemConf table (above).
}

Set the VC index and the starting time-slot of the SONET path in the
SONET extension table (pwVcCemSonetExtTable):
{
  pwVcCemSonetExtVcIndex      = 83
  pwVcCemSonetExtPathTimeSlot = 1
}

8 CEM MIB Definitions

PW-CEM-MIB DEFINITIONS ::= BEGIN

IMPORTS
  MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
  experimental, Integer32, Counter32, Unsigned32,
  Counter64
  FROM SNMPv2-SMI

  -- MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
  -- FROM SNMPv2-CONF

  TEXTUAL-CONVENTION, TruthValue, RowStatus, StorageType,
  TimeStamp
  FROM SNMPv2-TC
InterfaceIndex, InterfaceIndexOrZero
FROM IF-MIB

PwVcIndex
FROM PW-TC-MIB

PerfCurrentCount, PerfIntervalCount
FROM PerfHist-TC-MIB

pwVcIndex, pwVcInstance, pwVcOperStatus
FROM PW-MIB;

-- The CEM MIB

pwVcCemMIB MODULE-IDENTITY
LAST-UPDATED "200107191200Z"  -- 19 July 2001 12:00:00 EST
ORGANIZATION "Pseudo-Wire Emulation Edge-to-Edge (PWE3) Working Group"

CONTACT-INFO

  Dave Danenberg
  Postal: Litchfield Communications, Inc.
  76 Westbury Park Rd
  Princeton Building East
  Watertown, CT 06795
  Tel:    +1-860-945-1573 x3180
  Email:  dave_danenberg@litchfieldcomm.com

  Thomas D. Nadeau
  Postal: Cisco Systems, Inc.
  250 Apollo Drive
  Chelmsford, MA 01824
  Tel:    +1-978-244-3051
  Email:  tnadeau@cisco.com

  David Zelig

Danenberg et al Expires January 2002 8

Internet Draft PW CEM MIB July 2001

  Postal: Corrigent Systems LTD.
  126, Yigal Alon St.
  Tel Aviv, ISRAEL
  Phone: +972-3-6945273
  E-mail: davidz@corrigent.com

  Scott Park
  Postal: Litchfield Communications, Inc.
  76 Westbury Park Rd
  Princeton Building East
  Watertown, CT 06795
  Tel:    +1-860-945-1573 x3185
  Email:  scott_park@litchfieldcomm.com

  Andrew G. Malis
  Postal: Vivace Networks, Inc.
  2730 Orchard Parkway
  San Jose, CA 95134
  Email: Andy.Malis@vivacenetworks.com
DESCRIPTION


This MIB reports to the PW-MIB. The PW-MIB contains structures and MIB associations generic to Pseudo-Wire Virtual Circuit (VC) emulation. VC-specific MIBs (such as this) contain config and stats for specific VC types."

-- Revision history.

REVISION
"200107191200Z" -- 19 July 2001 12:00:00 EST
DESCRIPTION

"Based on the PWE3 Requirements/Framework, the original CEM MIB (draft-danenberg-sonet-ces-mpls-mib-00.txt) has been split into 3 drafts:

- PW-MIB (Zelig, et al). Service-independent MIB.
- PW-CEM-MIB (Danenberg, et al). Service-specific MIB.

Aside from the split, then PW-CEM-MIB has undergone other changes:

- Ability to define LOPS (loss of packet sync), SES
- Support for ‘unstructured’ operation.
- Support for CEM-RDI.
- Many more minor changes and clarifications."

::= { experimental 8888 } --To be assigned by IANA

-- Local Textual conventions
PwVcCemConfIndex ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION "Index into the pwVcCemConfTable. Removed trap OID declarations"
SYNTAX Unsigned32

-- Top level components of this MIB.

-- Traps
pwVcCemNotifications OBJECT IDENTIFIER ::= { pwVcCemMIB 0 }
pwVcCemNotifyPrefix OBJECT IDENTIFIER
::= { pwVcCemNotifications 0 }

-- Tables, Scalars
pwVcCemObjects OBJECT IDENTIFIER
::= { pwVcCemMIB 1 }

-- Conformance
-- pwVcCemConformance OBJECT IDENTIFIER
-- ::= { pwVcCemMIB 2 }

-- CEM VC table

pwVcCemTable OBJECT-TYPE
SYNTAX Sequence OF PwVcCemEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This table contains the index to the entry SONET path
table for this CEM VC, the index to the CEM config table,
and CEM statistics."
 ::= { pwVcCemObjects 1 }

pwVcCemEntry OBJECT-TYPE
SYNTAX PwVcCemEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This MIB reports to the PW-MIB. This table is indexed by
the same index that was created for the associated entry
in the PW VC Table (in the PW-MIB).

- The PwVcIndex.
- The PwVcInstance.

An entry is created in this table by the agent for every
entry in the pwVc table with a VcType of ‘cem’.

This table provides per CEM VC performance information.
HC (high capacity) counters are required for some counts
due to the high speeds expected with CEM services. A SONET
path of width 48 (STS-48c) can rollover non-HC counters in
a few minutes.

CEM bridges the SONET and packet worlds. In the packet
world, counts typically start from the time of service
creation - and don’t stop. In the SONET world, counts are
kept in 15 minute intervals. The CEM-MIB supports both
methods. Thestats in the VC CEM table count forever. The
15 minute interval counts are in the tables following this.

NOTE: Outbound errors are currently not considered in the
CEM-MIB. It is assumed that CEM packets are forwarded as
they are generated. i.e., It is assumed that there are no
congestion issues at the point of CEM adaptation."

INDEX { pwVcIndex, pwVcInstanceId }
::= { pwVcCemTable 1 }

PwVcCemEntry ::= SEQUENCE {

  pwVcCemType                  INTEGER,  
  pwVcCemSonetIfIndex          InterfaceIndexOrZero,  
  pwVcCemConfIndex             PwVcCemConfIndex,

  pwVcCemPerfDbaInPacketsHC    Counter64,  
  pwVcCemPerfDbaOutPacketsHC   Counter64,  
  pwVcCemPerfDbaInPackets      Counter32,  
  pwVcCemPerfDbaOutPackets     Counter32,

  pwVcCemPerfInNegPtrAdjust    Counter32,  
  pwVcCemPerfInPosPtrAdjust    Counter32,  
  pwVcCemPerfInPtrAdjustSecs   Counter32,  
  pwVcCemPerfOutNegPtrAdjust   Counter32,  
  pwVcCemPerfOutPosPtrAdjust   Counter32,  
  pwVcCemPerfOutPtrAdjustSecs  Counter32,  
  pwVcCemPerfAbsPtrAdjust      Integer32,  

  pwVcCemPerfCorrectableHdrErrors Counter32,  
  pwVcCemPerfUncorrectableHdrErrors Counter32,  
  pwVcCemPerfMissingPkts       Counter32,  
  pwVcCemPerfPktsOoseq         Counter32,  
  pwVcCemPerfPktsOoseqDropped  Counter32,  
  pwVcCemPerfJtrBfrUnderruns   Counter32,  
  pwVcCemPerfJtrBfrOverruns    Counter32,  
  pwVcCemPerfPktsBadLength     Counter32,

  pwVcCemPerfESs                Counter32,  
  pwVcCemPerfSESs               Counter32,  
  pwVcCemPerfUASs               Counter32,  
  pwVcCemPerfLastDefectsAtFailure BITS,  
  pwVcCemPerfLastFailureTimeStamp TimeStamp,  
  pwVcCemPerfDiscontinuityTime  TimeStamp

}

pwVcCemType OBJECT-TYPE
SYNTAX INTEGER {
  structured (1),
  unstructured (2)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION "Specifies if this CEM VC will carry a particular SONET path (structured type) or the entire SONET frame (unstructured type)."
DEFVAL { structured }
::= { pwVcCemEntry 1 }

pwVcCemSonetIfIndex OBJECT-TYPE
SYNTAX InterfaceIndexOrZero
MAX-ACCESS read-write
STATUS current
DESCRIPTION
"This is a unique index within the ifTable. It represents the interface index for the SONET path (if CEM type is structured) or SONET section/line (if unstructured). A value of zero indicates an interface index that has yet to be determined. Once set, if the SONET ifIndex is (for some reason) later removed, the agent MAY delete the associated PW VC rows (e.g., this pwVcCem table entry). If the agent does not delete the rows, it is recommended that the agent set this object to zero.

Note: if there are multiple instances of CEM VC, then the pathIfIndex object MUST be identical in each instance."

::= { pwVcCemEntry 2 }

pwVcCemConfIndex OBJECT-TYPE
SYNTAX        PwVcCemConfIndex
MAX-ACCESS    read-write
STATUS        current
DESCRIPTION
"Index to CEM configuration table below. It is likely that multiple CEM VCs will share 1 CEM conf table entry."

::= { pwVcCemEntry 3 }

pwVcCemPerfDbaInPacketsHC OBJECT-TYPE
SYNTAX        Counter64
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of DBA packets received."

::= { pwVcCemEntry 4 }

pwVcCemPerfDbaOutPacketsHC OBJECT-TYPE
SYNTAX        Counter64
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of DBA packets sent."

::= { pwVcCemEntry 5 }

pwVcCemPerfDbaInPackets OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of DBA packets received."

::= { pwVcCemEntry 6 }

pwVcCemPerfDbaOutPackets OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of DBA packets sent."

::= { pwVcCemEntry 7 }
-- Pointer adjustment stats

pwVcCemPerfInNegPtrAdjust OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Number of negative pointer adjustments sent on the
                SONET path based on CEM pointer adjustments received."
 ::= { pwVcCemEntry 8 }

pwVcCemPerfInPosPtrAdjust OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Number of positive pointer adjustments sent on the
                SONET path based on CEM pointer adjustments received."
 ::= { pwVcCemEntry 9 }

pwVcCemPerfInPtrAdjustSecs OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Number of seconds in which a pos or neg pointer
                adjustment was sent on the SONET path."
 ::= { pwVcCemEntry 10 }

pwVcCemPerfOutNegPtrAdjust OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Number of negative pointer adjustments seen on the
                SONET path and encoded onto sent CEM packets."
 ::= { pwVcCemEntry 11 }

pwVcCemPerfOutPosPtrAdjust OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Number of positive pointer adjustments seen on the
                SONET path and encoded onto sent CEM packets."
 ::= { pwVcCemEntry 12 }

pwVcCemPerfOutPtrAdjustSecs OBJECT-TYPE
SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Number of seconds in which a pos or neg pointer
                adjustment was seen on the SONET path."
 ::= { pwVcCemEntry 13 }

pwVcCemPerfAbsPtrAdjust OBJECT-TYPE
SYNTAX        Integer32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Absolute Pointer adjustments is relative adjustment
drifts between inbound and outbound streams. It is
calculated as such :
  ( InPosPtrAdjust  -   InNegPtrAdjust) -
   (OutPosPtrAdjust  -  OutNegPtrAdjust) "
::= { pwVcCemEntry 14 }

pwVcCemPerfCorrectableHdrErrors OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of correctable CEM header errors detected on
inbound CEM packets."
::= { pwVcCemEntry 15 }

pwVcCemPerfUncorrectableHdrErrors OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of uncorrectable CEM header errors detected on
inbound CEM packets."
::= { pwVcCemEntry 16 }

pwVcCemPerfMissingPkt OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of missing packets (as detected via CEM header
sequence number gaps)."
::= { pwVcCemEntry 17 }

pwVcCemPerfPktOoseq OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of packets detected out of sequence (via CEM
header sequence numbers), but were able to be re-sequenced.
Note: some implementations may not support this feature
(see pwVcCemConfPktResequence)."
::= { pwVcCemEntry 18 }

pwVcCemPerfPktOoseqDropped OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
"Number of packets detected out of sequence (via CEM
header sequence numbers), but were unable to be re-
sequenced.
::= { pwVcCemEntry 19 }

pwVcCemPerfJtrBfrUnderruns OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Number of times the jitter buffer transitioned to empty."
::= { pwVcCemEntry 20 }

pwVcCemPerfJtrBfrOverruns OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Number of packets received while jitter buffer is full."
::= { pwVcCemEntry 21 }

pwVcCemPerfPktsBadLength OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only

DESCRIPTION
"Number of packets received larger or smaller than
pwVcCemConfPktLength."
::= { pwVcCemEntry 22 }

pwVcCemPerfESs OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The counter associated with the number of Errored
Seconds encountered. See future versions of
draft-malis-sonet-ces-mpls for definition of ES."
::= { pwVcCemEntry 23 }

pwVcCemPerfSESs OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The counter associated with the number of Severely
Errored Seconds. See pwVcCemConfMissingPktsToSes.
Also see future versions of draft-malis-sonet-ces-mpls
for definition."
::= { pwVcCemEntry 24 }

pwVcCemPerfUASs OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The counter associated with the number of
UnAvailable Seconds. See pwVcCemConfSesToUAS."
NOTE: When first entering the UAS state, the number of SesToUas is added to this object, then as each additional UAS occurs, this object increments by one.

NOTE: Similar to RFC 2558, if the agent chooses to update the various performance statistics in real time then it must be prepared to retroactively reduce the ES, SES, counts (by the value of pwVcCemConfSesToUas) and increase the UAS count (by that same value) when it determines that UAS state has been entered.

::= { pwVcCemEntry 25 }

pwVcCemPerfLastDefectsAtFailure OBJECT-TYPE
SYNTAX BITS {
  uncrtHdr  ( 0),
  missingPkt ( 1),
  ooSeqDropped( 2),
  jtrBfrUnder ( 3),
  jtrBfrOver  ( 4),
  pktBadLength( 5),
  lops       ( 6),
  rdi        ( 7)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The state of defects when CEM VC last entered the a 'failure' state. Since more than one defect can contributed to a CEM VC failure, all are shown here.

Note: currently there is no hierarchy of CEM defects.

Note: the algorithm used to capture these defects when entering failure state is implementation specific.

Note: definition of 'CEM failure' is for further study. One possible definition: in LOPS state for 2.5 seconds. A 'CEM failure' may be also be determined (via software) as a result of trend analysis (e.g., missing packet ratio over time)."

::= { pwVcCemEntry 26 }

pwVcCemPerfLastFailureTimeStamp OBJECT-TYPE
SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The value of sysUpTime on the most recent occasion at which the CEM entered the 'failure' state."

::= { pwVcCemEntry 27 }

pwVcCemPerfDiscontinuityTime 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 or Counter64 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."

::= { pwVcCemEntry 28 }

-- End of PW CEM VC table

-- Obtain index for PW CEM Configuration table entries

pwVcCemConfIndexNext OBJECT-TYPE
SYNTAX PwVcCemConfIndex
MAX-ACCESS read-only
STATUS current
DESCRIPTION

Danenberg et al Expires January 2002 17

Internet Draft PW CEM MIB July 2001

"This object contains an appropriate value to be used for pwVcCemConfIndex when creating entries in the pwVcCemConfTable. The value 0 indicates that no unassigned entries are available. To obtain the value of pwVcCemConfIndex for a new entry in the pwVcCemConfTable, the manager issues a management protocol retrieval operation to obtain the current value of pwVcCemConfIndex. 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."

::= { pwVcCemObjects 2 }

-- PW CEM VC Configuration Table

pwVcCemConfTable OBJECT-TYPE
SYNTAX SEQUENCE OF PwVcCemConfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION

"This table contains a set of parameters that may be referenced by 1 or more CEM VCs by pwVcCemTable."

::= { pwVcCemObjects 3 }

pwVcCemConfEntry OBJECT-TYPE
SYNTAX PwVcCemConfEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION

"These parameters define the characteristics of a CEM VC. They are grouped here to ease NMS burden. Once an entry is created here it may be re-used by many VCs."

INDEX { pwVcCemConfTableIndex }
::= { pwVcCemConfTable 1 }

PwVcCemConfEntry ::= SEQUENCE {
  pwVcCemConfTableIndex PwVcCemConfIndex,
  pwVcCemConfigError BITS,
  pwVcCemConfMaxPktLength Unsigned32,
  pwVcCemConfMinPktLength Unsigned32,
  pwVcCemConfCemOptions Unsigned32,
  pwVcCemConfPktResequence TruthValue,
  pwVcCemConfEnableECC TruthValue,
  pwVcCemConfEnableDBA BITS,
  pwVcCemConfJtrBfrDepth Unsigned32,
  pwVcCemConfConsecPktsInsync Unsigned32,
  pwVcCemConfConsecMissingOutSync Unsigned32,
  pwVcCemConfErrorAction INTEGER,
  pwVcCemConfUserErrorAction Unsigned32,
  pwVcCemConfMissingPktsToSes Unsigned32,
  pwVcCemConfSesToUasUnsigned32,
  pwVcCemConfUasAction INTEGER,
  pwVcCemConfUserUasAction Unsigned32,
  pwVcCemConfApsCriteria BITS,
  pwVcCemConfRowStatus RowStatus,
  pwVcCemConfStorageType StorageType
}

pwVcCemConfTableIndex OBJECT-TYPE
SYNTAX PwVcCemConfIndex
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "Primary index to this table"
::= { pwVcCemConfEntry 1 }

pwVcCemConfigError OBJECT-TYPE
SYNTAX BITS {
  dba ( 0 ),
  resequencing ( 1 ),
  lopsPktCount ( 2 ),
  pktLength ( 3 ),
  jtrBfrDepth ( 4 ),
  cemOptions ( 5 ),
  ecc ( 6 ),
  timeslot ( 7 )
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Various configurations errors. Illegal settings within the pwVcCemConf table."
pwVcCemConfMaxPktLength OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"This is the maximum CEM packet length in number of bytes (including CEM header and payload). It is fixed and applies to inbound and outbound packets carrying user payload. Although there is no control over inbound packets, those of illegal length are discarded and accounted for (see pwVcCemPerfPktsBadLength.)

Used by signaling (LDP) to fill in the 'Virtual Circuit FEC element' and the CEM interface parameters therein."
REFERENCE

DEFVAL { 0 }

pwVcCemConfMinPktLength OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"This is the minimum CEM packet length in number of bytes (including CEM header and payload). It applies to CEM's bandwidth-savings packets. Currently DBA is the only bandwidth-savings packet type (in the future, CEM may support compression). Minimum packet length is necessary in some systems or networks.

Setting Zero here indicates that there is no minimum packet restriction. If non-zero, then packets too small are be discarded and accounted for (see pwVcCemPerfPktsBadLength.)

Used by signaling (LDP) to fill in the 'Virtual Circuit FEC element' and the CEM interface parameters therein."
REFERENCE

DEFVAL { 0 }
pwVcCemConfCemOptions OBJECT-TYPE
SYNTAX        Unsigned32
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
"Used by signaling (LDP) to fill in the 'Virtual Circuit FEC element' and the CEM interface parameters therein. CEM options are formatted into bit fields within a 16-bit word. Such fields will define CEM-DBA, STS-n starting time slot, etc."

REFERENCE
"See FEC element format and Interfaces Parameters format in Martini. L, et al, <draft-martini-l2circuit-trans-mpls-06>. Also see Malis, A., et al, <draft-malis-sonet-ces-mpls-05>."

pwVcCemConfPktResequence OBJECT-TYPE
SYNTAX        TruthValue
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
"If set True: as inbound packets are queued in the jitter buffer, out of order packets are re-sequenced. The maximum sequence number differential (i.e., the range in which resequencing can occur) is dependant on the depth of the jitter buffer. See pwVcCemConfJtrBfrDepth.

NOTE: Some implementations may not support this feature. If set True, then the 'resequencing' error code is set in pwVcCemConfigError.

NOTE: Some implementations may always have this feature on. If set False, then the 'resequencing' error code is set in pwVcCemConfigError."

pwVcCemConfEnableECC OBJECT-TYPE
SYNTAX        TruthValue
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
"If set True: As inbound packets are received, the CEM ECC is evaluated. If the CEM header is OK (or can be corrected, then the packet is process - else dropped. For outbound packets, ECC is calculated and inserted in the CEM header. If set false: Inbound ECC is ignored, outbound ECC is set to zero.

NOTE: Some implementations may not support ECC. If set True, then the 'ecc' error code is set in pwVcCemConfigError.

NOTE: Some implementations may always use ECC. If set False, then the 'ecc' error code is set in pwVcCemConfigError."
pwVcCemConfEnableDBA OBJECT-TYPE
SYNTAX BITS {
    allOnesOnAis(0),
    allZerosOnUnequipped(1)
}
MAX-ACCESS     read-create
STATUS        current
DESCRIPTION

"Any bits set here MUST enable the DBA (dynamic bandwidth allocation) feature for the specified condition. Setting allOnesOnAis will cause CEM packet payload suppression when AIS is detected on the associated SONET path. Similarly, allZerosOnUnequipped will cause payload suppression when the SONET path is un-equipped. During these conditions, CEM packets will continue to be sent, but with indicators set in the CEM header instructing the remote to play all ones or zeros onto its SONET path.

NOTE: some implementations may not support this feature. If set, then the 'dba' error code is set in pwVcCemConfigError."
DEFVAL { { allOnesOnAis } }
 ::= { pwVcCemConfEntry 8 }

pwVcCemConfJtrBfrDepth OBJECT-TYPE
SYNTAX        Unsigned32
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
"This setting configures the number of packet buffers reserved for this CEM VC. This object essentially sets the maximum amount of time allowed between CEM packets before the jitter buffer empties. This variable should be set based on the SONET path width (speed) and the amount of delay variation expected to be introduced by the network. Like bandwidth, jitter buffers are likely to be a limited resource to be managed."
 ::= { pwVcCemConfEntry 9 }

--
-- The following counters work together to integrate (filter) errors and the lack of errors on the CEM VC. An error is caused by a missing packet. Missing packet can be a results of: packet out of sequence (uncorrectable), CEM header error (uncorrectable), pkt length error, jitter buffer overflow, and jitter buffer underflow. The result is declaring whether or not the CEM VC is in Loss of Packet Sync (LOPS) state.
--

pwVcCemConfConsecPktsInsync OBJECT-TYPE
SYNTAX        Unsigned32
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
"Consecutive pkts with sequential sequence numbers required to exit the LOPS state."
REFERENCE
DEFVAL { 2 }
::= { pwVcCemConfEntry 10 }

pwVcCemConfConsecMissingOutSync OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create

Danenberg et al Expires January 2002 22
Internet Draft PW CEM MIB July 2001

STATUS current
DESCRIPTION "Consecutive missing pkts required to enter
the LOPS state."
DEFVAL { 10 }
::= { pwVcCemConfEntry 11 }

pwVcCemConfErrorAction OBJECT-TYPE
SYNTAX INTEGER {
    playAllOnes (1),
    userValue   (2)
}
MAX-ACCESS read-create
STATUS current
DESCRIPTION "These are the actions to take when inbound packets are
missing due to gap in sequence numbers (uncorrectable via
available re-sequencing), jitter buffer underruns, or
packets with bad CEM headers. These patterns are sent
played) on the SONET path. These settings are used for
immediate errors and (unless the pwVcCemConfUasAction is
'errorAction') are not in effect once the CEM VC is in
UAS state. For userValue see pwVcCemConfUserErrorAction."
DEFVAL { playAllOnes }
::= { pwVcCemConfEntry 12 }

pwVcCemConfUserErrorAction OBJECT-TYPE
SYNTAX Unsigned32 (0..255)
MAX-ACCESS read-create
STATUS current
DESCRIPTION "Used when userValue selected above. This byte value is
sent on every byte in every SPE."
DEFVAL { 255 } -- Play all ones
::= { pwVcCemConfEntry 13 }

pwVcCemConfMissingPktsToSes 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 }
::= { pwVcCemConfEntry 14 }
"Number of consecutive SESs before declaring VC in UAS state (and at which point pwVcCemPerfUASs starts counting). The SesToUas default value is 10 seconds.

NOTE: Similar to RFC 2558, If the agent chooses to update the various performance statistics in real time then it must be prepared to retroactively reduce the ES, SES, counts by this value and increase the UAS count by this value when it determines that UAS state has been entered.

NOTE: 10 consecutive non-SESs cause the VC to exit the UAS state. See pwVcCemPerfSESs and pwVcCemPerfUASs."

DEFVAL { 10 }
::= { pwVcCemConfEntry 15 }

"These are the actions to take once the CEM VC has been declared in UAS state (as observed in pwVcCemPerfDefects) and there are missing packets (as described in the pwVcCemConfErrorAction object, above).
These patterns are sent (played) on the SONET path. If 'errorAction' is selected, then the pwVcCemConfErrorAction settings stay in effect even after the CEM VC is in UAS.
For userValue see pwVcCemConfUserUasAction."

DEFVAL { playAllOnes }
::= { pwVcCemConfEntry 16 }

"Used when userValue selected above. This byte value is sent on every byte in every SPE."

DEFVAL { 255 } -- Play all ones
::= { pwVcCemConfEntry 17 }
Bits set here represent defects used as inputs into APS mechanisms (see PW-MIB).
Note: for APS applications requiring fast response, the 'lops' trigger option is recommended. LOPS can be generated within milliseconds, while UAS is multiple seconds.
Note: ‘rdi’ is the CEM Remote Defect Indicator and is a result of the remote end being in LOPS state.

DEFVAL { { lops } }

::= { pwVcCemConfEntry 18 }

pwVcCemConfRowStatus  OBJECT-TYPE
SYNTAX                 RowStatus
MAX-ACCESS             read-create
STATUS                 current
DESCRIPTION            "For creating, modifying, and deleting this row."
::= { pwVcCemConfEntry 19 }

pwVcCemConfStorageType  OBJECT-TYPE
SYNTAX                  StorageType
MAX-ACCESS              read-create
STATUS                  current
DESCRIPTION             "This variable indicates the storage type for this object."
::= { pwVcCemConfEntry 20 }

-- End of PW CEM VC Configuration Parameter Table

-- CEM VC Performance Current Interval Table.

pwVcCemPerfCurrentTable  OBJECT-TYPE
SYNTAX                  SEQUENCE OF PwVcCemPerfCurrentEntry
MAX-ACCESS              not-accessible
STATUS                  current
DESCRIPTION             "This table provides per CEM VC performance information much like the pwVcCemPerfTable above. However, these counts represent the current 15 minute interval only. NOTE: Counter64 objects are used here, Counter32 is too small for OC-768 CEM VCs."
::= { pwVcCemObjects 4 }

pwVcCemPerfCurrentEntry  OBJECT-TYPE
SYNTAX                  PwVcCemPerfCurrentEntry
MAX-ACCESS              not-accessible
STATUS                  current
DESCRIPTION             "An entry in this table is created by the agent for every
pwVcCemConf entry. It is an extension to the
pwVcCemConf table. After 15 minutes, the contents of
this table entry are copied to a new entry in the
pwVcCemPerfInterval table and the counts in this entry are
reset to zero.

AUGMENTS { pwVcCemEntry }

::= { pwVcCemPerfCurrentTable 1 }

PwVcCemPerfCurrentEntry ::= SEQUENCE {
  pwVcCemPerfCurrentDbaInPacketsHC     Counter64,
  pwVcCemPerfCurrentDbaOutPacketsHC    Counter64,
  pwVcCemPerfCurrentDbaInPackets       Counter32,
  pwVcCemPerfCurrentDbaOutPackets      Counter32,

  pwVcCemPerfCurrentInNegPtrAdjust     Counter32,
  pwVcCemPerfCurrentInPosPtrAdjust     Counter32,
  pwVcCemPerfCurrentInPtrAdjustSecs    Counter32,
  pwVcCemPerfCurrentOutNegPtrAdjust    Counter32,
  pwVcCemPerfCurrentOutPosPtrAdjust    Counter32,
  pwVcCemPerfCurrentOutPtrAdjustSecs   Counter32,
  pwVcCemPerfCurrentAbsPtrAdjust       Integer32,

  pwVcCemPerfCurrentCrctHdrErrors      Counter32,
  pwVcCemPerfCurrentUncrctHdrErrors    Counter32,
  pwVcCemPerfCurrentMissingPkts        Counter32,
  pwVcCemPerfCurrentPktsOoseq          Counter32,
  pwVcCemPerfCurrentPktsOoseqDropped   Counter32,
  pwVcCemPerfCurrentJtrBfrUnderruns    Counter32,
  pwVcCemPerfCurrentJtrBfrOverruns     Counter32,
  pwVcCemPerfCurrentPktsBadLength      Counter32,

  pwVcCemPerfCurrentESs                PerfCurrentCount,
  pwVcCemPerfCurrentSESs               PerfCurrentCount,
  pwVcCemPerfCurrentUASs               PerfCurrentCount
}

 pwVcCemPerfCurrentDbaInPacketsHC OBJECT-TYPE
 SYNTAX Counter64
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION "Number of DBA packets received."
 ::= { pwVcCemPerfCurrentEntry 1 }

 pwVcCemPerfCurrentDbaOutPacketsHC OBJECT-TYPE
 SYNTAX Counter64
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION "Number of DBA packets sent."
 ::= { pwVcCemPerfCurrentEntry 2 }

 pwVcCemPerfCurrentDbaInPackets OBJECT-TYPE
 SYNTAX Counter32
pwVcCemPerfCurrentDbaOutPackets

--- Pointer adjustment stats

dwVcCemPerfCurrentInNegPtrAdjust

dwVcCemPerfCurrentInPosPtrAdjust

dwVcCemPerfCurrentInPtrAdjustSecs

dwVcCemPerfCurrentOutNegPtrAdjust

dwVcCemPerfCurrentOutPosPtrAdjust
Internet Draft             PW CEM MIB                     July 2001

STATUS        current
DESCRIPTION
   "Number of positive pointer adjustments seen on the
   SONET path and encoded onto sent CEM packets."
::= { pwVcCemPerfCurrentEntry 9 }

pwVcCemPerfCurrentOutPtrAdjustSecs OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of seconds in which a pos or neg pointer
   adjustment was seen on the SONET path."
::= { pwVcCemPerfCurrentEntry 10 }

pwVcCemPerfCurrentAbsPtrAdjust OBJECT-TYPE
SYNTAX        Integer32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Absolute Pointer adjustments is relative adjustment
   drifts between inbound and outbound streams. It is
   calculated as such :
   ( InPosPtrAdjust - InNegPtrAdjust) - 
   (OutPosPtrAdjust - OutNegPtrAdjust) "
::= { pwVcCemPerfCurrentEntry 11 }

pwVcCemPerfCurrentCrctHdrErrors OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of correctable CEM header errors detected on
   inbound CEM packets."
::= { pwVcCemPerfCurrentEntry 12 }

pwVcCemPerfCurrentUncrctHdrErrors OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of uncorrectable CEM header errors detected on
   inbound CEM packets."
::= { pwVcCemPerfCurrentEntry 13 }

pwVcCemPerfCurrentMissingPkts OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of missing packets (as detected via CEM header
   sequence number gaps)."
::= { pwVcCemPerfCurrentEntry 14 }

Danenberg et al       Expires January 2002                       28
Internet Draft             PW CEM MIB                     July 2001
pwVcCemPerfCurrentPktsOoseq OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of packets detected out of sequence (via CEM
   header sequence numbers), but were able to be re-sequenced.
   Note: some implementations may not support this
   feature (see pwVcCemConfPktResequence)."
::= { pwVcCemPerfCurrentEntry 15 }

pwVcCemPerfCurrentPktsOoseqDropped OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of packets detected out of sequence (via CEM
   header sequence numbers), but were unable to be re-
   sequenced."
::= { pwVcCemPerfCurrentEntry 16 }

pwVcCemPerfCurrentJtrBfrUnderruns OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of times the jitter buffer transitioned to empty."
::= { pwVcCemPerfCurrentEntry 17 }

pwVcCemPerfCurrentJtrBfrOverruns OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of packets received while jitter buffer is full."
::= { pwVcCemPerfCurrentEntry 18 }

pwVcCemPerfCurrentPktsBadLength OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "Number of non-bandwidth-saving packets received larger
   or smaller than pwVcCemConfMaxPktLength. Plus number of
   bandwidth-saving packets (like DBA) that are smaller
   than pwVcCemConfMinPktLength."
::= { pwVcCemPerfCurrentEntry 19 }

pwVcCemPerfCurrentESs OBJECT-TYPE
SYNTAX        PerfCurrentCount
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
   "The counter associated with the number of Errored
   Seconds encountered."
::= { pwVcCemPerfCurrentEntry 20 }

pwVcCemPerfCurrentSESs OBJECT-TYPE
SYNTAX        PerfCurrentCount
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
 "The counter associated with the number of
  Severely Errored Seconds encountered."
::= { pwVcCemPerfCurrentEntry 21 }

pwVcCemPerfCurrentUASs OBJECT-TYPE
SYNTAX        PerfCurrentCount
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
 "The counter associated with the number of
  Unavailable Seconds encountered."
::= { pwVcCemPerfCurrentEntry 22 }

-- End PW CEM VC Performance Current Interval Table

-- PW CEM VC Performance Interval Table.

pwVcCemPerfIntervalTable OBJECT-TYPE
SYNTAX        SEQUENCE OF PwVcCemPerfIntervalEntry
MAX-ACCESS    not-accessible
STATUS        current
DESCRIPTION
 "This table provides per CEM VC performance information
  much like the pwVcCemPerfCurrentTable 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.
  NOTE: Counter64 objects are used here, Counter32 is
  too small for OC-768 CEM VCs."
::= { pwVcCemObjects 5 }

pwVcCemPerfIntervalEntry OBJECT-TYPE
SYNTAX        PwVcCemPerfIntervalEntry
MAX-ACCESS    not-accessible
STATUS        current
DESCRIPTION
 "An entry in this table is created by the agent for
  every pwVcCemPerfCurrentEntry 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.
  pwVcCemIndex is found in the pwVcCemConf table."

INDEX  { pwVcIndex, pwVcInstance,
      pwVcCemPerfIntervalNumber }

Danenberg et al       Expires January 2002                       30
Internet Draft             PW CEM MIB                     July 2001

::= { pwVcCemPerfIntervalTable 1 }
PwVcCemPerfIntervalEntry ::= SEQUENCE {
  pwVcCemPerfIntervalNumber             Unsigned32,
  pwVcCemPerfIntervalValidData          TruthValue,
  pwVcCemPerfIntervalReset              INTEGER,
  pwVcCemPerfIntervalDbaInPacketsHC     Counter64,
  pwVcCemPerfIntervalDbaOutPacketsHC    Counter64,
  pwVcCemPerfIntervalDbaInPackets       Counter32,
  pwVcCemPerfIntervalDbaOutPackets      Counter32,
  pwVcCemPerfIntervalInNegPtrAdjust     Counter32,
  pwVcCemPerfIntervalInPosPtrAdjust     Counter32,
  pwVcCemPerfIntervalInPtrAdjustSecs    Counter32,
  pwVcCemPerfIntervalOutNegPtrAdjust    Counter32,
  pwVcCemPerfIntervalOutPosPtrAdjust    Counter32,
  pwVcCemPerfIntervalOutPtrAdjustSecs   Counter32,
  pwVcCemPerfIntervalAbsPtrAdjust       Integer32,
  pwVcCemPerfIntervalCrctHdrErrors      Counter32,
  pwVcCemPerfIntervalUncrctHdrErrors    Counter32,
  pwVcCemPerfIntervalMissingPkts        Counter32,
  pwVcCemPerfIntervalPktsOoseq          Counter32,
  pwVcCemPerfIntervalPktsOoseqDropped   Counter32,
  pwVcCemPerfIntervalJtrBfrUnderruns    Counter32,
  pwVcCemPerfIntervalJtrBfrOverruns     Counter32,
  pwVcCemPerfIntervalPktsBadLength      Counter32,
  pwVcCemPerfIntervalESs                PerfIntervalCount,
  pwVcCemPerfIntervalSESs               PerfIntervalCount,
  pwVcCemPerfIntervalUASs               PerfIntervalCount
}

pwVcCemPerfIntervalNumber 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."
  ::= { pwVcCemPerfIntervalEntry 1 }

pwVcCemPerfIntervalValidData OBJECT-TYPE
SYNTAX        TruthValue
MAX-ACCESS    read-only
STATUS        current

DESCRIPTION
  "This variable indicates if the data for this interval is valid."
  ::= { pwVcCemPerfIntervalEntry 2 }
pwVcCemPerfIntervalReset OBJECT-TYPE
  SYNTAX        INTEGER {
    reset (1),
    normal(2)
  }
  MAX-ACCESS    read-create
  STATUS        current
  DESCRIPTION
    "Used in cases where the user knows that the errors
    within this interval should not be counted. Writing
    'reset' sets all error counts to zero."
  ::= { pwVcCemPerfIntervalEntry 3 }

pwVcCemPerfIntervalDbaInPacketsHC OBJECT-TYPE
  SYNTAX        Counter64
  MAX-ACCESS    read-only
  STATUS        current
  DESCRIPTION
    "Number of DBA packets received."
  ::= { pwVcCemPerfIntervalEntry 4 }

pwVcCemPerfIntervalDbaOutPacketsHC OBJECT-TYPE
  SYNTAX        Counter64
  MAX-ACCESS    read-only
  STATUS        current
  DESCRIPTION
    "Number of DBA packets sent."
  ::= { pwVcCemPerfIntervalEntry 5 }

pwVcCemPerfIntervalDbaInPackets OBJECT-TYPE
  SYNTAX        Counter32
  MAX-ACCESS    read-only
  STATUS        current
  DESCRIPTION
    "Number of DBA packets received."
  ::= { pwVcCemPerfIntervalEntry 6 }

pwVcCemPerfIntervalDbaOutPackets OBJECT-TYPE
  SYNTAX        Counter32
  MAX-ACCESS    read-only
  STATUS        current
  DESCRIPTION
    "Number of DBA packets sent."
  ::= { pwVcCemPerfIntervalEntry 7 }

-- Pointer adjustment stats
pwVcCemPerfIntervalInNegPtrAdjust OBJECT-TYPE
  SYNTAX        Counter32
  MAX-ACCESS    read-only
  STATUS        current
  DESCRIPTION
    "Number of negative pointer adjustments sent on the
    SONET path based on CEM pointer adjustments received."
  ::= { pwVcCemPerfIntervalEntry 8 }

pwVcCemPerfIntervalInPosPtrAdjust OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Number of postive pointer adjustments sent on the
SONET path based on CEM pointer adjustments received."
 ::= { pwVcCemPerfIntervalEntry 9 }

pwVcCemPerfIntervalInPtrAdjustSecs OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Number of seconds in which a pos or neg pointer
adjustment was sent on the SONET path."
 ::= { pwVcCemPerfIntervalEntry 10 }

pwVcCemPerfIntervalOutNegPtrAdjust OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Number of negative pointer adjustments seen on the
SONET path and encoded onto sent CEM packets."
 ::= { pwVcCemPerfIntervalEntry 11 }

pwVcCemPerfIntervalOutPosPtrAdjust OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Number of positive pointer adjustments seen on the
SONET path and encoded onto sent CEM packets."
 ::= { pwVcCemPerfIntervalEntry 12 }

pwVcCemPerfIntervalOutPtrAdjustSecs OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION   "Number of seconds in which a pos or neg pointer
adjustment was seen on the SONET path."
 ::= { pwVcCemPerfIntervalEntry 13 }

pwVcCemPerfIntervalAbsPtrAdjust OBJECT-TYPE
SYNTAX        Integer32
MAX-ACCESS    read-only

DESCRIPTION   "Absolute Pointer adjustments is relative adjustment
drifts between inbound and outbound streams. It is calculated as such :
( InPosPtrAdjust - InNegPtrAdjust ) -
( OutPosPtrAdjust - OutNegPtrAdjust ) "
 ::= { pwVcCemPerfIntervalEntry 14 }
pwVcCemPerfIntervalCrctHdrErrors OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION    "Number of correctable CEM header errors detected on inbound CEM packets."
 ::= { pwVcCemPerfIntervalEntry 15 }

pwVcCemPerfIntervalUncrctHdrErrors OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION    "Number of uncorrectable CEM header errors detected on inbound CEM packets."
 ::= { pwVcCemPerfIntervalEntry 16 }

pwVcCemPerfIntervalMissingPkts OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION    "Number of missing packets (as detected via CEM header sequence number gaps)."
 ::= { pwVcCemPerfIntervalEntry 17 }

pwVcCemPerfIntervalPktsOoseq OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION    "Number of packets detected out of sequence (via CEM header sequence numbers), but were able to be re-sequenced. Note: some implementations may not support this feature (see pwVcCemConfPktResequence)."
 ::= { pwVcCemPerfIntervalEntry 18 }

pwVcCemPerfIntervalPktsOoseqDropped OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION    "Number of packets detected out of sequence (via CEM header sequence numbers), but were unable to be re-sequenced."  
 ::= { pwVcCemPerfIntervalEntry 19 }

pwVcCemPerfIntervalJtrBfrUnderruns OBJECT-TYPE
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION    "Number of times the jitter buffer transitioned to empty."  
 ::= { pwVcCemPerfIntervalEntry 20 }
pwVcCemPerfIntervalJtrBfrOverruns OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Number of packets received while jitter buffer is full."
::= { pwVcCemPerfIntervalEntry 21 }

pwVcCemPerfIntervalPktsBadLength OBJECT-TYPE
SYNTAX Counter32
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Number of packets received larger or smaller than pwVcCemConfPktLength."
::= { pwVcCemPerfIntervalEntry 22 }

pwVcCemPerfIntervalESs OBJECT-TYPE
SYNTAX PerfIntervalCount
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The counter associated with the number of Errored Seconds encountered."
::= { pwVcCemPerfIntervalEntry 23 }

pwVcCemPerfIntervalSESs OBJECT-TYPE
SYNTAX PerfIntervalCount
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The counter associated with the number of Severely Errored Seconds encountered."
::= { pwVcCemPerfIntervalEntry 24 }

pwVcCemPerfIntervalUASs OBJECT-TYPE
SYNTAX PerfIntervalCount
MAX-ACCESS read-only
STATUS current
DESCRIPTION "The counter associated with the number of Unavailable Seconds encountered."
::= { pwVcCemPerfIntervalEntry 25 }

-- End PW CEM VC Performance Interval Table

-- PW CEM SONET Path or Section/Line Extension Table.

pwVcCemSonetExtTable OBJECT-TYPE
SYNTAX SEQUENCE OF PwVcCemSonetExtEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "This table associates a SONET path (for structured)
or section/line (for unstructured) with a CEM VC (or a set of CEM VCs) and provides the starting time-slot of the SONET path. It is assumed that the transmit and receive sides of the SONET path have the same starting time-slots and sonetPathCurrentWidth.

NOTE: It is assumed that a SONET path has standard time-slot distribution. For example, an OC-12 SONET line containing a path of width 3 (STS-3c) and starting at time-slot 1, will consume time-slots 1, 5, and 9.

::= { pwVcCemObjects 6 }

pwVcCemSonetExtEntry OBJECT-TYPE
SYNTAX          PwVcCemSonetExtEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
"This table is indexed by the SONET ifIndex. An entry to this table is created by the agent when an entry (intended for CEM) is created in either the sonetPathCurrentTable or sonetLineCurrentTable. Alternatively, an agent may create an entry in this table at the point where a valid ifIndex is written to pwVcCemSonetIfIndex in the pwVcCem table. The pwVcCemSonetExt table binds a SONET path (or section/line) to a PW CEM VC. A VC index of 0 indicates that the binding has not yet been set."

INDEX  { pwVcCemSonetExtIfIndex }
::= { pwVcCemSonetExtTable 1 }

PwVcCemSonetExtEntry ::= SEQUENCE {
    pwVcCemSonetExtIfIndex             InterfaceIndex,
    pwVcCemSonetExtVcIndex             PwVcIndex,
    pwVcCemSonetExtTimeSlot            Unsigned32,
    pwVcCemSonetExtStorageType         StorageType
}

pwVcCemSonetExtIfIndex  OBJECT-TYPE
SYNTAX        InterfaceIndex
MAX-ACCESS    not-accessible
STATUS        current
DESCRIPTION
"Index for a row in the sonetPathCurrentTable or sonetLineCurrentTable."
REFERENCE
"See ‘Definitions of Managed Objects for the SONET/SDH Interface Type’. Tesink, K., RFC 2558."
::= { pwVcCemSonetExtEntry 1 }

pwVcCemSonetExtVcIndex  OBJECT-TYPE
SYNTAX        PwVcIndex
MAX-ACCESS    read-write
STATUS        current
DESCRIPTION

"Index for the conceptual row identifying a VC (or set of VCs) within the generic PW VC table (in PW-MIB) or the local pwVcCemTable."

::= { pwVcCemSonetExtEntry 2 }

pwVcCemSonetExtTimeSlot OBJECT-TYPE
SYNTAX Unsigned32 (1..192)
MAX-ACCESS read-write
STATUS current
DESCRIPTION

"In structured CEM, this object indicates the starting time-slot for this SONET path within the SONET line and section. For OC-48, this value could range from 1 to 48. The SONET path width must be taken into consideration here. For example, in an OC-48, an STS-3c could not start at time-slot 47.

NOTE: In unstructured CEM, this object must be zero."
REFERENCE "RFC 2558, sonetPathCurrentWidth."

::= { pwVcCemSonetExtEntry 3 }

pwVcCemSonetExtStorageType OBJECT-TYPE
SYNTAX StorageType
MAX-ACCESS read-write
STATUS current
DESCRIPTION

"This variable indicates the storage type for this object."

::= { pwVcCemSonetExtEntry 4 }

-- End of CEM SONET Path or Section/Line Extension Table.

-- Notifications - CEM VC

pwVcCemGlobalNotifyEnable OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-write
STATUS current
DESCRIPTION

"If this object is true, then it enables the generation of pwVcCemUp and pwVcCemDown notifications, otherwise these notifications are not emitted.

Note: enabling/disabling notifications for individual CEM VCs is for further study."

DEFVAL { false }

::= { pwVcCemObjects 8 }

pwVcCemUp NOTIFICATION-TYPE
OBJECTS { pwVcOperStatus }
STATUS current
DESCRIPTION

"This notification is generated when a CEM VC has exited the 'failure' state."
pwVcCemNotifyPrefix 1

pwVcCemDown NOTIFICATION-TYPE
OBJECTS { pwVcCemPerfLastDefectsAtFailure }
STATUS current
DESCRIPTION
"This notification is generated when a CEM VC has entered the 'failure' state. See description of pwVcCemPerfLastDefectsAtFailure object."

END

9 References


Danenberg et al Expires January 2002

Internet Draft PW CEM MIB July 2001


[SONETMIB] Tesink, K. "Definitions of Managed objects for the SONET/SDH Interface Type", RFC 2558.


Danenberg et al       Expires January 2002                       40
Internet Draft     PW CEM MIB       July 2001


10 Author’s Addresses

Dave Danenberg
Litchfield Communications, Inc.
76 Westbury Park Rd
Princeton Building East
Watertown, CT 06795
Email: dave_danenberg@litchfieldcomm.com

David Zelig
Corrigent Systems LTD.
126, Yigal Alon st.
Tel Aviv, ISRAEL
Phone: +972-3-6945273
Email: davidz@corrigent.com

Danenberg et al Expires January 2002

Internet Draft PW CEM MIB July 2001

Thomas D. Nadeau
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
Email: tnadeau@cisco.com

Andrew G. Malis
Vivace Networks, Inc.
2730 Orchard Parkway
San Jose, CA 95134
Email: Andy.Malis@vivacenetworks.com

Scott C. Park
Litchfield Communications, Inc.
76 Westbury Park Rd
Princeton Building East
Watertown, CT 06795
Email: scott_park@litchfieldcomm.com

11 Full Copyright Statement

"Copyright (C) The Internet Society (2001). All Rights Reserved. This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it
or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.