Network Working Group
Request for Comments: 5066
Category: Standards Track

E. Beili
Actelis Networks

November 2007

Ethernet in the First Mile Copper (EFMCu) Interfaces MIB

Status of This Memo

Abstract

This document defines Management Information Base (MIB) modules for use with network management protocols in TCP/IP-based internets. This document describes extensions to the Ethernet-like Interfaces MIB and Medium Attachment Unit (MAU) MIB modules with a set of objects for managing Ethernet in the First Mile Copper (EFMCu) interfaces 10PASS-TS and 2BASE-TL, defined in IEEE Std 802.3ah-2004 (note: IEEE Std 802.3ah-2004 has been integrated into IEEE Std 802.3-2005). In addition, a set of objects is defined, describing cross-connect capability of a managed device with multi-layer (stacked) interfaces, extending the stack management objects in the Interfaces Group MIB and the Inverted Stack Table MIB modules.

Table of Contents

1. Introduction

New Ethernet-like interfaces have been defined in the Institute of Electrical and Electronics Engineers (IEEE) Standard 802.3ah-2004 [802.3ah], a.k.a. Ethernet in the First Mile (EFM), which is now a part of the base IEEE Standard 802.3-2005 [802.3]. In particular, 2BASE-TL and 10PASS-TS physical interfaces (PHYs), defined over voice-grade copper pairs, have been specified for the long and short reach, respectively. These interfaces, collectively called EFM Copper (EFMCu), are based on Single-pair High-speed Digital Subscriber Line (SHDSL) [G.991.2] and Very High speed Digital Subscriber Line (VDSL) [G.993.1] technology, supporting optional Physical Medium Entity (PME) aggregation (a.k.a. multi-pair bonding) with variable rates.

2BASE-TL PHY is capable of providing at least 2 Mbps over a 2700 m long single copper pair with a mean Bit Error Rate (BER) of 10^-7 (using 5 dB target noise margin).

10PASS-TS PHY is capable of providing at least 10 Mbps over a 750 m long single copper pair with a mean BER of 10^-7 (using 6 dB target noise margin).

This memo defines a Management Information Base (MIB) module for use with network management protocols in the Internet community to manage EFMCu interfaces. In addition, a MIB module is defined describing the cross-connect capability of a stacked interface.

Note that managed objects for Operation, Administration and Maintenance (OAM) and Ethernet over Passive Optical Networks (EPON) clauses of IEEE 802.3ah are defined in EFM-COMMON-MIB [RFC4878] and EFM-EPON-MIB [RFC4837], respectively.

2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies MIB modules that are compliant to the SMIv2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

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 [RFC2119].

3. Relation to Other MIB Modules

This section outlines the relationship of the MIB modules defined in this document with other MIB modules described in the relevant RFCs. Specifically, the Interfaces Group MIB (IF-MIB), Ethernet-Like (EtherLike-MIB), MAU (MAU-MIB), SHDSL (HDSL2-SHDSL-LINE-MIB), and VDSL (VDSL-LINE-EXT-MCM-MIB) modules are discussed.

3.1. Relation to Interfaces Group MIB Module

2BASE-TL and 10PASS-TS PHYs specified in the EFM-CU-MIB module are stacked (a.k.a. aggregated or bonded) Ethernet interfaces and as such are managed using generic interface management objects defined in the IF-MIB [RFC2863].

The stack management (i.e., actual connection of the sub-layers to the top-layer interface) is done via the ifStackTable, as defined in the IF-MIB [RFC2863], and its inverse ifInvStackTable, as defined in the IF-INVERTED-STACK-MIB [RFC2864].

The new tables ifCapStackTable and its inverse ifInvCapStackTable defined in the IF-CAP-STACK-MIB module below, extend the stack management with an ability to describe possible connections or cross-connect capability, when a flexible cross-connect matrix is present between the interface layers.

3.1.1. Layering Model

An EFMCu interface can aggregate up to 32 Physical Medium Entity (PME) sub-layer devices (modems), using the so-called PME Aggregation Function (PAF).

A generic EFMCu device can have a number of Physical Coding Sublayer (PCS) ports, each connected to a Media Access Controller (MAC) via a Medium Independent Interface (MII) at the upper layer, and cross-connected to a number of underlying PMEs, with a single PCS per PME relationship. See clause 61.1 of [802.3ah] for more details.

Each PME in the aggregated EFMCu port is represented in the Interface table (ifTable) as a separate interface with ifType of shdsl(169) for 2BASE-TL or vdsl(97) for 10PASS-TS. The ifType values are defined in [IANAifType-MIB].

ifSpeed for each PME SHALL return the actual data bitrate of the active PME (e.g., for 2BaseTL PMEs it is a multiple of 64 Kbps). A zero value SHALL be returned when the PME is Initializing or Down.

The ifSpeed of the PCS is the sum of the current operating data rates of all PMEs in the aggregation group, without the 64/65-octet encapsulation overhead and PAF overhead, but accounting for the Inter-Frame Gaps (IFGs).

When using the stated definition of ifSpeed for the PCS, there would be no frame loss in the following configuration (the test-sets are configured to generate 100% of back-to-back traffic, i.e., minimal IFG, at 10 or 100 Mbps, with min and max frame sizes; the EFM interfaces are aggregated, to achieve the shown speed):

```
      .-------.           .--.           .---.           .-------.
      |testset|--10BaseT--|CO|--2BaseTL--|CPE|--10BaseT--|testset|
      '-------'           '--'           '---'           '-------'
        ifSpeed= 10 Mbps        10 Mbps         10 Mbps

      .-------.            .--.            .---.            .-------.
      |testset|--100BaseT--|CO|--10PassTS--|CPE|--100BaseT--|testset|
      '-------'            '--'            '---'            '-------'
        ifSpeed= 100 Mbps        100 Mbps         100 Mbps
```

Figure 1: Example configuration with no frame loss

The following figure shows the IEEE 802.3 layering diagram and corresponding use of ifTable and ifMauTable:

```
 .-------------------------.  -
 |        LLC              |  ^
 +-------------------------+  | 1 ifEntry
 |        MAC              |  |     ifType: ethernetCsmacd(6)
 +-------------------------+  )   ifMauEntry
 |        Reconsiliation   |  |     ifMauType: dot3MauType2BaseTL or
 +-------------------------+  |                dot3MauType10PassTS
 |        PCS              |  v
 +-------------+---+-------+  -
 | TC \        |   |       |  ^
 +-----\       |   |       |  |
 | PMA  )PME 1 |...| PME N |  ) N ifEntry  (N=1..32)
 +-----/       |   |       |  |     ifType: shdsl(169) or vdsl(97)
 | PMD/        |   |       |  v
 ’-------------+---+-------’  -

  LLC - Logical Link Control       PMA - Physical Medium Attachment
  MAC - Media Access Control       PMD - Physical Medium Dependent
  PCS - Physical Coding Sub-layer  PME - Physical Medium Entity
  TC  - Transmission Convergence
```

Figure 2: Use of ifTable and ifMauTable for EFMCu ports

The ifStackTable is indexed by the ifIndex values of the aggregated EFMCu port (PCS) and the PMEs connected to it. ifStackTable allows a Network Management application to determine which PMEs are connected to a particular PCS and change connections (if supported by the application). The ifInvStackTable, being an inverted version of the ifStackTable, provides an efficient means for a Network Management application to read a subset of the ifStackTable and thereby determine which PCS runs on top of a particular PME.

A new table ifCapStackTable, defined in the IF-CAP-STACK-MIB module, specifies for each higher-layer interface (e.g., PCS port) a list of lower-layer interfaces (e.g., PMEs), which can possibly be cross-connected to that higher-layer interface, determined by the cross-connect capability of the device. This table, modeled after ifStackTable, is read-only, reflecting current cross-connect capability of stacked interface, which can be dynamic in some implementations (e.g., if PMEs are located on a pluggable module and the module is pulled out). Note that PME availability per PCS, described by ifCapStackTable, can be constrained by other parameters, for example, by aggregation capacity of a PCS or by the PME in question being already connected to another PCS. So, in order to ensure that a particular PME can be connected to the PCS, all respective parameters (e.g., ifCapStackTable, ifStackTable, and efmCuPAFCapacity) SHALL be inspected.

The ifInvCapStackTable, also defined in the IF-CAP-STACK-MIB module, describes which higher-layer interfaces (e.g., PCS ports) can possibly be connected to a particular lower-layer interface (e.g., PME), providing an inverted mapping of the ifCapStackTable. While it contains no additional information beyond that already contained in the ifCapStackTable, the ifInvCapStackTable has the ifIndex values in its INDEX clause in the reverse order, i.e., the lower-layer interface first, and the higher-layer interface second, providing an efficient means for a Network Management application to read a subset of the ifCapStackTable and thereby determine which interfaces can be connected to run on top of a particular interface.

3.1.2. PME Aggregation Function (PAF)

The PME Aggregation Function (PAF) allows a number of PMEs to be aggregated onto a PCS port, by fragmenting the Ethernet frames, transmitting the fragments over multiple PMEs, and assembling the original frames at the remote port. PAF is OPTIONAL, meaning that a device with a single PME MAY perform fragmentation and re-assembly if this function is supported by the device. Note however that the agent is REQUIRED to report on the PAF capability for all EFMCu ports (2BASE-TL and 10PASS-TS).

The EFM-CU-MIB module allows a Network Management application to query the PAF capability and enable/disable it if supported. Note that enabling PAF effectively turns on fragmentation and re-assembly, even on a single-PME port.

3.1.3. Discovery Operation

The EFMCu ports may optionally support discovery operation, whereby PMEs, during initialization, exchange information about their respective aggregation groups (PCS). This information can then be used to detect copper misconnections or for an automatic assignment of the local PMEs into aggregation groups instead of a fixed pre-configuration.

The MIB modules defined in this document allow a Network Management application to control the EFM Discovery mechanism and query its results. Note that the Discovery mechanism can work only if PAF is supported and enabled.

Two tables are used by the EFM Discovery mechanism: ifStackTable and ifCapStackTable. The following pseudo-code gives an example of the Discovery and automatic PME assignment for a generic PAF-enabled multi-PCS EFMCu device, located at Central Office (CO), using objects defined in these MIB modules and in the IF-MIB (Note that automatic PME assignment is only shown here for the purposes of the example. Fixed PME pre-assignment, manual assignment, or auto-assignment using an alternative internal algorithm may be chosen by a particular implementation):

```
// Go over all PCS ports in the CO device
FOREACH pcs[i] IN CO_device
{ // Perform discovery and auto-assignment only on PAF enabled ports
  // with room for more PMEs
  IF ( pcs[i].PAFSupported AND pcs[i].NumPMEs &lt; pcs[i].PAFCapacity )
  { // Assign a unique 6-octet local discovery code to the PCS
    // e.g., MAC address
    dc = pcs[i].DiscoveryCode = MAC[i];
    // Go over all disconnected PMEs, which can
    // potentially be connected to the PCS
    FOREACH pme[j] IN ifCapStackTable[pcs[i]] AND
                    NOT IN ifStackTable[pcs[i]]  // not connected
    { // Try to grab the remote RT_device, by writing the value
      // of the local 6-octet discovery code to the remote
      // discovery code register (via handshake mechanism).
      // This operation is atomic Set-if-Clear action, i.e., it
      // would succeed only if the remote discovery register was
      // zero.  Read the remote discovery code register via Get
      // operation to see if the RT_device, attached via the PME
      // is indeed marked as being the CO_device peer.
      pme[j].RemoteDiscoveryCode = dc;          // Set-if-Clear
      r = pme[j].RemoteDiscoveryCode;           // Get
      IF ( r == dc AND pcs[i].NumPMEs &lt; pcs[i].PAFCapacity)
      { // Remote RT_device connected via PME[j] is/was a peer
        // for PCS[i] and there is room for another PME in the
        // PCS[i] aggregation group (max. PAF capacity is not
        // reached yet).
        // Connect this PME to the PCS (via ifStackTable,
        // ifInvStackTable being inverse of ifStackTable is
        // updated automatically, i.e., pcs[i] is auto-added
        // to ifInvStackTable[pme[j]])
        ADD pme[j] TO ifStackTable[pcs[i]];
        pcs[i].NumPMEs = pcs[i].NumPMEs + 1;
        // Discover all other disconnected PMEs,
        // attached to the same RT_device and connect them to
        // the PCS provided there is enough room for more PMEs.
        FOREACH pme[k] IN ifCapStackTable[pcs[i]] AND
                        NOT IN ifStackTable[pcs[i]]
```

```
         { // Get Remote Discovery Code from the PME to see if
           // it belongs to a connected RT_device "grabbed" by
           // the CO_device.
           r = pme[k].RemoteDiscoveryCode;
           IF ( r == dc AND pcs[i].NumPMEs &lt; pcs[i].PAFCapacity)
           { // Physically connect the PME to the PCS
             // (pcs[i] is auto-added TO ifInvStackTable[pme[k]])
             ADD pme[k] TO ifStackTable[pcs[i]];
             pcs[i].NumPMEs = pcs[i].NumPMEs + 1;
           }
         }
       }
       // At this point we have discovered all local PMEs which
       // are physically connected to the same remote RT_device
       // and connected them to PCS[i].  Go to the next PCS.
       BREAK;
     }
   }
 }
```

An SNMP Agent for an EFMCu device builds the ifCapStackTable and its inverse ifInvCapStackTable according to the information contained in the Clause 45 PME_Available_register (see [802.3ah] 61.1.5.3 and 45.2.3.20).

Adding a PME to the ifStackTable row for a specific PCS involves actual connection of the PME to the PCS, which can be done by modifying Clause 45 PME_Aggregate_register (see [802.3ah] 61.1.5.3 and 45.2.3.21).

Note that the PCS port does not have to be operationally 'down' for the connection to succeed. In fact, a dynamic PME addition (and removal) MAY be implemented with an available PME being initialized first (by setting its ifAdminStatus to 'up') and then added to an operationally 'up' PCS port, by modifying a respective ifStackTable (and respective ifInvStackTable) entry.

It is RECOMMENDED that a removal of the last operationally 'up' PME from an operationally 'up' PCS would be rejected by the implementation, as this action would completely drop the link.

3.1.4. EFMCu Ports Initialization

EFMCu ports being built on top of xDSL technology require a lengthy initialization or 'training' process, before any data can pass. During this initialization, both ends of a link (peers) work cooperatively to achieve the required data rate on a particular copper pair. Sometimes, when the copper line is too long or the noise on the line is too high, that 'training' process may fail to achieve a specific target rate with required characteristics.

The ifAdminStatus object from the IF-MIB controls the desired state of a PCS with all the PMEs connected to it or of an individual PME port. Setting this object to 'up' instructs a particular PCS or PME to start the initialization process, which may take tens of seconds for EFMCu ports, especially if PAF is involved. The ifOperStatus object shows the operational state of an interface (extended by the ifMauMediaAvailable object from MAU-MIB for PCS and efmCuPmeOperStatus defined in the EFM-CU-MIB module for PME interfaces).

A disconnected PME may be initialized by changing the ifAdminState from 'down' to 'up'. Changing the ifAdminState to 'up' on the PCS initializes all PMEs connected to that particular PCS. Note that in case of PAF some interfaces may fail to initialize while others succeed. The PCS is considered operationally 'up' if at least one PME aggregated by its PAF is operationally 'up'. When all PMEs connected to the PCS are 'down', the PCS SHALL be considered operationally 'lowerLayerDown'. The PCS SHALL be considered operationally 'notPresent' if it is not connected to any PME. The PCS/PME interface SHALL remain operationally 'down' during initialization.

The efmCuPmeOperStatus defined in the EFM-CU-MIB module expands PME's ifOperStatus value of 'down' to 'downReady', 'downNotReady', and 'init' values, indicating various EFMCu PME-specific states.

3.1.5. Usage of ifTable

Both PME and PCS interfaces of the EFMCu PHY are managed using interface-specific management objects defined in the EFM-CU-MIB module and generic interface objects from the ifTable of IF-MIB, with all management table entries referenced by the interface index ifIndex.

The following table summarizes EFMCu-specific interpretations for some of the ifTable objects specified in the mandatory ifGeneralInformationGroup:

IF-MIB object	EFMCu interpretation
ifIndex	Interface index. Note that each PME and each PCS in the EFMCu PHY MUST have a unique index, as there are some PCS- and PME-specific attributes accessible only on the PCS or PME level.
ifType	ethernetCsmacd(6) for PCS, shdsl(169) for 2BASE-TL PME, vdsl(97) for 10PASS-TS PME.
ifSpeed	Operating data rate for the PME. For the PCS, it is the sum of the current operating data rates of all PMEs in the aggregation group, without the 64/65-octet encapsulation overhead and PAF overhead, but accounting for the Inter-Frame Gaps (IFGs).
ifAdminStatus	Setting this object to ’up’ instructs a particular PCS (with all PMEs connected to it) or PME to start initialization process.
ifOperStatus	efmCuPmeOperStatus supplements the ’down’ value of ifOperStatus for PMEs.

Table 1: EFMCu interpretation of IF-MIB objects

3.2. Relation to SHDSL MIB Module

G.SHDSL.bis modems, similar to PMEs comprising a 2BASE-TL port, are described in the HDSL2-SHDSL-LINE-MIB module [RFC4319]. Note that not all attributes of G.SHDSL modems reflected in the HDSL2-SHDSL-LINE-MIB module have adequate management objects (Clause 30 attributes and Clause 45 registers) in the EFM standard.

Because of these differences and for the purposes of simplicity, unification of attributes common to both 2BASE-TL and 10PASS-TS PMEs, and name consistency (e.g., prefixing the 2BASE-TL PME related objects with ’efmCuPme2B’ instead of ’hdsl2shdsl’), it was decided not to reference HDSL2-SHDSL-LINE-MIB objects, but define all the relevant objects in the EFM-CU-MIB module.

However, if some functionality not available in the EFM-CU-MIB module is required and supported by the PME, e.g., performance monitoring, relevant HDSL2-SHDSL-LINE-MIB groups MAY be included and applied for PMEs of 2BASE-TL subtype.

3.3. Relation to VDSL MIB Module

VDSL modems, similar to the PME(s) comprising a 10PASS-TS port, are described in the VDSL-LINE-EXT-MCM-MIB module [RFC4070]. Note that not all attributes of VDSL modems reflected in the VDSL-LINE-EXT-MCM-MIB module have adequate management objects (Clause 30 attributes and Clause 45 registers) in the EFM standard.

Because of these differences and for the purposes of simplicity, unification of attributes common to both 2BASE-TL and 10PASS-TS PMEs, and name consistency, it was decided not to reference VDSL-LINE-EXT-MCM-MIB objects, but define all the relevant objects in the EFM-CU-MIB module.

However, if some functionality not available in the EFM-CU-MIB module is required and supported by the PME, relevant VDSL-LINE-EXT-MCM-MIB groups MAY be included and applied for PMEs of 10PASS-TS subtype.

3.4. Relation to Ethernet-Like and MAU MIB Modules

The implementation of the EtherLike-MIB [RFC3635] and MAU-MIB [RFC4836] modules is REQUIRED for EFMCu interfaces.

Two new values of ifMauType (OBJECT-IDENTITIES of dot3MauType) and corresponding bit definitions of ifMauTypeListBits (IANAifMauTypeListBits) have been defined in the IANA-MAU-MIB module [RFC4836] for EFMCu MAUs:

- dot3MauType2BaseTL and b2BaseTL - for 2BASE-TL MAU

- dot3MauType10PassTS and b10PassTS - for 10PASS-TS MAU

Additionally, the IANA-MAU-MIB module defines two new values of ifMauMediaAvailable, specifically for EFMCu ports: availableReduced and ready (in textual convention IANAifMauMediaAvailable). Due to the PME aggregation, the EFMCu interpretation of some possible ifMauMediaAvailable values differs from other MAUs as follows:

- unknown - the EFMCu interface (PCS with connected PMEs) is Initializing

- ready - the interface is Down, at least one PME in the aggregation group (all PMEs connected to the PCS) is ready for handshake

- available - the interface is Up, all PMEs in the aggregation group are up

- notAvailable - the interface is Down, all PMEs in the aggregation group are Down, no handshake tones are detected by any PME

- availableReduced - the interface is Up, a link fault is detected at the receive direction by one or more PMEs in the aggregation group, but at least one PME is Up

- pmdLinkFault - a link fault is detected at the receive direction by all PMEs in the aggregation group

As an EtherLike interface, every EFMCu port (an ifEntry representing a consolidation of LLC, MAC, and PCS (sub)layers) SHALL return an ifType of ethernetCsmacd(6). While most of the MAU characteristics are not applicable to the EFMCu ports (no auto-negotiation, false carriers, or jabber), they SHALL return an appropriate ifMauType (dot3MauType2BaseTL or dot3mauType10PassTS) in order to direct the management software to look in the EFM-CU-MIB module for the desired information. For example, the information on the particular EFMCu flavor that an EFMCu port is running is available from efmCuOperSubType, defined in the EFM-CU-MIB module.

Since EFMCu PMEs are not EtherLike interfaces, they cannot be instantiated as MAU interface objects.

4. MIB Structure

4.1. EFM Copper MIB Overview

The main management objects defined in the EFM-CU-MIB module are split into 2 groups:

- efmCuPort - containing objects for configuration, capabilities, status, and notifications, common to all EFMCu PHYs.

- efmCuPme - containing objects for configuration, capabilities, status, and notifications of EFMCu PMEs.

The efmCuPme group in turn contains efmCuPme2B and efmCuPme10P groups, which define PME profiles specific to 2BASE-TL and 10PASS-TS PMEs, respectively, as well as PME-specific status information.

4.2. Interface Stack Capability MIB Overview

The IF-CAP-STACK-MIB module contains 2 tables:

- ifCapStackTable - containing objects that define possible relationships among the sub-layers of an interface with flexible cross-connect (cross-connect capability).

- ifInvCapStackTable - an inverse of the ifCapstackTable.

4.3. PME Profiles

Since a managed node can have a large number of EFMCu PHYs, provisioning every parameter on every EFMCu PHY may become burdensome. Moreover, most PMEs are provisioned identically with the same set of parameters. To simplify the provisioning process, the EFM-CU-MIB module makes use of configuration profiles, similar to the HDSL2-SHDSL-LINE-MIB and VDSL-LINE-EXT-MCM-MIB modules. A profile is a set of parameters, used either for configuration or representation of a PME. The same profile can be shared by multiple PME ports using the same configuration.

The PME profiles are defined in the efmCuPme2BProfileTable and efmCuPme10PProfileTable for 2BASE-TL and 10PASS-TS PMEs, respectively. There are 12 predefined standard profiles for 2BASE-TL and 22 standard profiles for 10PASS-TS, defined in 802.3ah and dedicated for rapid provisioning of EFMCu PHYs in most scenarios. In addition, the EFM-CU-MIB defines two additional predefined profiles for "best-effort" provisioning of 2BASE-TL PMEs. An ability to define new configuration profiles is also provided to allow for EFMCu deployment tailored to specific copper environments and spectral regulations.

A specific configuration or administrative profile is assigned to a specific PME via the efmCuPmeAdminProfile object. If efmCuPmeAdminProfile is zero, then the efmCuAdminProfile object of the PCS port connected to the PME determines the configuration profile (or a list of possible profiles) for that PME. This mechanism allows specifying a common profile for all PMEs connected to the PCS port, with an ability to change individual PME profiles by setting efmCuPmeAdminProfile object, which overwrites the profile set by efmCuAdminProfile.

A current operating PME profile is pointed to by the efmCuPmeOperProfile object. Note that this profile entry can be created automatically to reflect achieved parameters in adaptive (not fixed) initialization.

4.4. Mapping of IEEE 802.3ah Managed Objects

This section contains the mapping between relevant managed objects (attributes) defined in [802.3ah] Clause 30, and managed objects defined in this document and in associated MIB modules, i.e., the IF-MIB [RFC2863].

Note that the majority of the objects defined in the EFM-CU-MIB module do not have direct counterparts in Clause 30 and instead refer to Clause 45 registers.

IEEE 802.3 Managed Object	Corresponding SNMP Object
oMAU - Basic Package (Mandatory)	
aMAUType	ifMauType (MAU-MIB)
aMAUTypeList	ifMauTypeListBits (MAU-MIB)
aMediaAvailable	ifMediaAvailable (MAU-MIB)
oPAF - Basic Package (Mandatory)	
aPAFID	ifIndex (IF-MIB)
aPhyEnd	efmCuPhySide
aPHYCurrentStatus	efmCuStatus
aPAFSupported	efmCuPAFSupported
oPAF - PME Aggregation Package (Optional)	
aPAFAdminState	efmCuPAFAdminState
aLocalPAFCapacity	efmCuPAFCapacity
aLocalPMEAvailable	ifCapStackTable
aLocalPMEAggregate	ifStackTable (IF-MIB)
aRemotePAFSupported	efmCuRemotePAFSupported
aRemotePAFCapacity	efmCuRemotePAFCapacity
aRemotePMEAggregate	
oPME - 10P/2B Package (Mandatory)	
aPMEID	ifIndex (IF-MIB)

aPMEAdminState	ifAdminState (IF-MIB)
aPMEStatus aPMESNRMgn	efmCuPmeStatus efmCuPmeSnrMgn
aTCCodingViolations	efmCuPmeTCCodingErrors
aTCCRCErrors	efmCuPmeTCCrcErrors
aProfileSelect	efmCuAdminProfile, efmCuPmeAdminProfile
aOperatingProfile	efmCuPmeOperProfile
aPMEFECCorrectedBlocks	efmCuPme10PFECCorrectedBlocks
aPMEFECUncorrectableBlocks	efmCuPme10PFECUncorrectedBlocks

Table 2: Mapping of IEEE 802.3 Managed Objects

5. Interface Stack Capability MIB Definitions

```
IF-CAP-STACK-MIB DEFINITIONS ::= BEGIN

  IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, mib-2
      FROM SNMPv2-SMI         -- [RFC2578]
    TruthValue
      FROM SNMPv2-TC          -- [RFC2579]
    MODULE-COMPLIANCE, OBJECT-GROUP
      FROM SNMPv2-CONF        -- [RFC2580]
    ifStackGroup2, ifStackHigherLayer, ifStackLowerLayer
      FROM IF-MIB             -- [RFC2863]
    ifInvStackGroup
      FROM IF-INVERTED-STACK-MIB -- [RFC2864]
    ;

  ifCapStackMIB MODULE-IDENTITY
    LAST-UPDATED "200711070000Z"  -- November 07, 2007
    ORGANIZATION "IETF Ethernet Interfaces and Hub MIB Working Group"
    CONTACT-INFO
      "WG charter:
        http://www.ietf.org/html.charters/OLD/hubmib-charter.html

      Mailing Lists:
        General Discussion: hubmib@ietf.org
```

```
           To Subscribe: hubmib-request@ietf.org
           In Body: subscribe your_email_address

         Chair:  Bert Wijnen
         Postal: Alcatel-Lucent
                 Schagen 33
                 3461 GL Linschoten
                 Netherlands
          Phone: +31-348-407-775
          EMail: bwijnen@alcatel-lucent.com

         Editor: Edward Beili
         Postal: Actelis Networks Inc.
                 25 Bazel St., P.O.B. 10173
                 Petach-Tikva 10173
                 Israel
          Phone: +972-3-924-3491
          EMail: edward.beili@actelis.com"

       DESCRIPTION
         "The objects in this MIB module are used to describe
         cross-connect capabilities of stacked (layered) interfaces,
         complementing ifStackTable and ifInvStackTable defined in
         IF-MIB and IF-INVERTED-STACK-MIB, respectively.

         Copyright (C) The IETF Trust (2007).  This version
         of this MIB module is part of RFC 5066;  see the RFC
         itself for full legal notices."

       REVISION    "200711070000Z"  -- November 07, 2007
       DESCRIPTION "Initial version, published as RFC 5066."

       ::= { mib-2 166 }

      -- Sections of the module
      -- Structured as recommended by [RFC4181], see
      -- Appendix D: Suggested OID Layout

      ifCapStackObjects     OBJECT IDENTIFIER ::= { ifCapStackMIB 1 }

      ifCapStackConformance OBJECT IDENTIFIER ::= { ifCapStackMIB 2 }

      -- Groups in the module

      --
      -- ifCapStackTable group
      --
      ifCapStackTable  OBJECT-TYPE
        SYNTAX      SEQUENCE OF IfCapStackEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "This table, modeled after ifStackTable from IF-MIB,
          contains information on the possible 'on-top-of'
          relationships between the multiple sub-layers of network
          interfaces (as opposed to actual relationships described in
          ifStackTable).  In particular, it contains information on
          which sub-layers MAY possibly run 'on top of' which other
          sub-layers, as determined by cross-connect capability of the
          device, where each sub-layer corresponds to a conceptual row
          in the ifTable.  For example, when the sub-layer with ifIndex
          value x can be connected to run on top of the sub-layer with
          ifIndex value y, then this table contains:

            ifCapStackStatus.x.y=true

          The ifCapStackStatus.x.y row does not exist if it is
          impossible to connect between the sub-layers x and y.

          Note that for most stacked interfaces (e.g., 2BASE-TL)
          there's always at least one higher-level interface (e.g., PCS
          port) for each lower-level interface (e.g., PME) and at
          least one lower-level interface for each higher-level
          interface, that is, there is at least a single row with a
          'true' status for any such existing value of x or y.

          This table is read-only as it describes device capabilities."
        REFERENCE
          "IF-MIB, ifStackTable"
        ::= { ifCapStackObjects 1 }

      ifCapStackEntry  OBJECT-TYPE
        SYNTAX      IfCapStackEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Information on a particular relationship between two
          sub-layers, specifying that one sub-layer MAY possibly run
          on 'top' of the other sub-layer.  Each sub-layer corresponds
          to a conceptual row in the ifTable (interface index for
          lower and higher layer, respectively)."
        INDEX {
          ifStackHigherLayer,
          ifStackLowerLayer
        }
        ::= { ifCapStackTable 1 }

      IfCapStackEntry ::= SEQUENCE {
           ifCapStackStatus       TruthValue
         }

      ifCapStackStatus  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The status of the 'cross-connect capability' relationship
          between two sub-layers.  The following values can be returned:
            true(1)         - indicates that the sub-layer interface,
                              identified by the ifStackLowerLayer MAY
                              be connected to run 'below' the sub-layer
                              interface, identified by the
                              ifStackHigherLayer index.
            false(2)        - the sub-layer interfaces cannot be
                              connected temporarily due to
                              unavailability of the interface(s), e.g.,
                              one of the interfaces is located on an
                              absent pluggable module.

          Note that lower-layer interface availability per higher-layer,
          indicated by the value of 'true', can be constrained by
          other parameters, for example, by the aggregation capacity of
          a higher-layer interface or by the lower-layer interface in
          question being already connected to another higher-layer
          interface.  In order to ensure that a particular sub-layer can
          be connected to another sub-layer, all respective objects
          (e.g., ifCapStackTable, ifStackTable, and efmCuPAFCapacity for
          EFMCu interfaces) SHALL be inspected.

          This object is read-only, unlike ifStackStatus, as it
          describes a cross-connect capability."
        ::= { ifCapStackEntry 1 }

      ifInvCapStackTable  OBJECT-TYPE
        SYNTAX        SEQUENCE OF IfInvCapStackEntry
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
          "A table containing information on the possible relationships
          between the multiple sub-layers of network interfaces.  This
          table, modeled after ifInvStackTable from
          IF-INVERTED-STACK-MIB, is an inverse of the ifCapStackTable
          defined in this MIB module.
          In particular, this table contains information on which
          sub-layers MAY run 'underneath' which other sub-layers, where
          each sub-layer corresponds to a conceptual row in the ifTable.
          For example, when the sub-layer with ifIndex value x MAY be
          connected to run underneath the sub-layer with ifIndex value
          y, then this table contains:

             ifInvCapStackStatus.x.y=true

          This table contains exactly the same number of rows as the
          ifCapStackTable, but the rows appear in a different order.

          This table is read-only as it describes a cross-connect
          capability."
        REFERENCE
           "IF-INVERTED-STACK-MIB, ifInvStackTable"
        ::= { ifCapStackObjects 2 }

      ifInvCapStackEntry  OBJECT-TYPE
        SYNTAX        IfInvCapStackEntry
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
           "Information on a particular relationship between two sub-
           layers, specifying that one sub-layer MAY run underneath the
           other sub-layer.  Each sub-layer corresponds to a conceptual
           row in the ifTable."
        INDEX { ifStackLowerLayer, ifStackHigherLayer }
        ::= { ifInvCapStackTable 1 }

       IfInvCapStackEntry ::= SEQUENCE {
         ifInvCapStackStatus       TruthValue
       }

      ifInvCapStackStatus  OBJECT-TYPE
        SYNTAX         TruthValue
        MAX-ACCESS     read-only
        STATUS         current
        DESCRIPTION
           "The status of the possible 'cross-connect capability'
           relationship between two sub-layers.

           An instance of this object exists for each instance of the
           ifCapStackStatus object, and vice versa.  For example, if the
           variable ifCapStackStatus.H.L exists, then the variable
           ifInvCapStackStatus.L.H must also exist, and vice versa.  In
           addition, the two variables always have the same value.
```

```
             The ifInvCapStackStatus object is read-only, as it describes
             a cross-connect capability."
          REFERENCE
             "ifCapStackStatus"
          ::= { ifInvCapStackEntry 1 }

     --
     -- Conformance Statements
     --

      ifCapStackGroups      OBJECT IDENTIFIER ::=
           { ifCapStackConformance 1 }

      ifCapStackCompliances OBJECT IDENTIFIER ::=
           { ifCapStackConformance 2 }

      -- Units of Conformance

      ifCapStackGroup OBJECT-GROUP
        OBJECTS {
          ifCapStackStatus,
          ifInvCapStackStatus
        }
        STATUS  current
        DESCRIPTION
          "A collection of objects providing information on the
          cross-connect capability of multi-layer (stacked) network
          interfaces."
        ::= { ifCapStackGroups 1 }


     -- Compliance Statements

      ifCapStackCompliance MODULE-COMPLIANCE
        STATUS      current
        DESCRIPTION
          "The compliance statement for SNMP entities, which provide
          information on the cross-connect capability of multi-layer
          (stacked) network interfaces, with flexible cross-connect
          between the sub-layers."


        MODULE  -- this module
          MANDATORY-GROUPS {
            ifCapStackGroup
          }

          OBJECT       ifCapStackStatus
          SYNTAX       TruthValue { true(1) }
          DESCRIPTION
            "Support for the false(2) value is OPTIONAL for
            implementations supporting pluggable interfaces."

          OBJECT       ifInvCapStackStatus
          SYNTAX       TruthValue { true(1) }
          DESCRIPTION
            "Support for the false(2) value is OPTIONAL for
            implementations supporting pluggable interfaces."

        MODULE  IF-MIB
          MANDATORY-GROUPS {
            ifStackGroup2
          }

        MODULE  IF-INVERTED-STACK-MIB
          MANDATORY-GROUPS {
            ifInvStackGroup
          }

        ::= { ifCapStackCompliances 1 }
   END
```

6. EFM Copper MIB Definitions

```
   EFM-CU-MIB DEFINITIONS ::= BEGIN

     IMPORTS
       MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE, Integer32,
       Unsigned32, Counter32, mib-2
         FROM SNMPv2-SMI         -- [RFC2578]
       TEXTUAL-CONVENTION, TruthValue, RowStatus, PhysAddress
         FROM SNMPv2-TC          -- [RFC2579]
       MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
         FROM SNMPv2-CONF        -- [RFC2580]
       SnmpAdminString
         FROM SNMP-FRAMEWORK-MIB -- [RFC3411]
       ifIndex, ifSpeed
         FROM IF-MIB             -- [RFC2863]
       ;

     efmCuMIB MODULE-IDENTITY
       LAST-UPDATED "200711140000Z"  -- November 14, 2007
       ORGANIZATION "IETF Ethernet Interfaces and Hub MIB Working Group"
       CONTACT-INFO
         "WG charter:
           http://www.ietf.org/html.charters/OLD/hubmib-charter.html
         Mailing Lists:
           General Discussion: hubmib@ietf.org
           To Subscribe: hubmib-request@ietf.org
           In Body: subscribe your_email_address

         Chair:  Bert Wijnen
         Postal: Alcatel-Lucent
                 Schagen 33
                 3461 GL Linschoten
                 Netherlands
          Phone: +31-348-407-775
          EMail: bwijnen@alcatel-lucent.com

         Editor: Edward Beili
         Postal: Actelis Networks Inc.
                 25 Bazel St., P.O.B. 10173
                 Petach-Tikva 10173
                 Israel
          Phone: +972-3-924-3491
          Email: edward.beili@actelis.com"

       DESCRIPTION
         "The objects in this MIB module are used to manage
         the Ethernet in the First Mile (EFM) Copper (EFMCu) Interfaces
         2BASE-TL and 10PASS-TS, defined in IEEE Std. 802.3ah-2004,
         which is now a part of IEEE Std. 802.3-2005.

         The following references are used throughout this MIB module:

         [802.3ah] refers to:
           IEEE Std 802.3ah-2004: 'IEEE Standard for Information
           technology - Telecommunications and information exchange
           between systems - Local and metropolitan area networks -
           Specific requirements -
           Part 3: Carrier Sense Multiple Access with Collision
           Detection (CSMA/CD) Access Method and Physical Layer
           Specifications -
           Amendment: Media Access Control Parameters, Physical
           Layers and Management Parameters for Subscriber Access
           Networks', 07 September 2004.

         Of particular interest are Clause 61, 'Physical Coding
         Sublayer (PCS) and common specifications, type 10PASS-TS and
         type 2BASE-TL', Clause 30, 'Management', Clause 45,
         'Management Data Input/Output (MDIO) Interface', Annex 62A,
         'PMD profiles for 10PASS-TS' and Annex 63A, 'PMD profiles for
         2BASE-TL'.
         [G.991.2] refers to:
           ITU-T Recommendation G.991.2: 'Single-pair High-speed Digital
           Subscriber Line (SHDSL) transceivers', December 2003.

         [ANFP] refers to:
           NICC Document ND1602:2005/08: 'Specification of the Access
           Network Frequency Plan (ANFP) applicable to transmission
           systems used on the BT Access Network,' August 2005.

         The following normative documents are quoted by the DESCRIPTION
         clauses in this MIB module:

         [G.993.1] refers to:
           ITU-T Recommendation G.993.1: 'Very High speed Digital
           Subscriber Line transceivers', June 2004.

         [T1.424] refers to:
           ANSI T1.424-2004: 'Interface Between Networks and Customer
           Installation Very-high-bit-rate Digital Subscriber Lines
           (VDSL) Metallic Interface (DMT Based)', June 2004.

         [TS 101 270-1] refers to:
           ETSI TS 101 270-1: 'Transmission and Multiplexing (TM);
           Access transmission systems on metallic access cables;
           Very high speed Digital Subscriber Line (VDSL); Part 1:
           Functional requirements', October 2005.

         Naming Conventions:
           Atn   - Attenuation
           CO    - Central Office
           CPE   - Customer Premises Equipment
           EFM   - Ethernet in the First Mile
           EFMCu - EFM Copper
           MDIO  - Management Data Input/Output
           Mgn   - Margin
           PAF   - PME Aggregation Function
           PBO   - Power Back-Off
           PCS   - Physical Coding Sublayer
           PMD   - Physical Medium Dependent
           PME   - Physical Medium Entity
           PSD   - Power Spectral Density
           SNR   - Signal to Noise Ratio
           TCPAM - Trellis Coded Pulse Amplitude Modulation

         Copyright (C) The IETF Trust (2007).  This version
         of this MIB module is part of RFC 5066;  see the RFC
         itself for full legal notices."
```

```
        REVISION    "200711140000Z"  -- November 14, 2007
        DESCRIPTION "Initial version, published as RFC 5066."

        ::= { mib-2 167 }

       -- Sections of the module

       efmCuObjects     OBJECT IDENTIFIER ::= { efmCuMIB 1 }

       efmCuConformance OBJECT IDENTIFIER ::= { efmCuMIB 2 }

       -- Groups in the module

       efmCuPort        OBJECT IDENTIFIER ::= { efmCuObjects 1 }

       efmCuPme         OBJECT IDENTIFIER ::= { efmCuObjects 2 }

       -- Textual Conventions

       EfmProfileIndex ::= TEXTUAL-CONVENTION
         DISPLAY-HINT "d"
         STATUS       current
         DESCRIPTION
           "A unique value, greater than zero, for each PME configuration
           profile in the managed EFMCu port.  It is RECOMMENDED that
           values are assigned contiguously starting from 1.  The value
           for each profile MUST remain constant at least from one
           re-initialization of the entity’s network management system
           to the next re-initialization."
         SYNTAX       Unsigned32 (1..255)

       EfmProfileIndexOrZero ::= TEXTUAL-CONVENTION
         DISPLAY-HINT "d"
         STATUS       current
         DESCRIPTION
           "This textual convention is an extension of the
           EfmProfileIndex convention.  The latter defines a greater than
           zero value used to identify a PME profile in the managed EFMCu
           port.  This extension permits the additional value of zero.
           The value of zero is object-specific and MUST therefore be
           defined as part of the description of any object that uses
           this syntax.
           Examples of the usage of zero value might include situations
           where the current operational profile is unknown."
         SYNTAX       Unsigned32 (0..255)

       EfmProfileIndexList ::= TEXTUAL-CONVENTION
         DISPLAY-HINT "1d:"
```

```
    STATUS       current
    DESCRIPTION
      "This textual convention represents a list of up to 6
      EfmProfileIndex values, any of which can be chosen for
      configuration of a PME in a managed EFMCu port.
      The EfmProfileIndex textual convention defines a greater than
      zero value used to identify a PME profile.
      The value of this object is a concatenation of zero or
      more (up to 6) octets, where each octet contains an 8-bit
      EfmProfileIndex value.
      A zero-length octet string is object-specific and MUST
      therefore be defined as part of the description of any object
      that uses this syntax.  Examples of the usage of a zero-length
      value might include situations where an object using this
      textual convention is irrelevant for a specific EFMCu port
      type."
    SYNTAX       OCTET STRING (SIZE(0..6))

  EfmTruthValueOrUnknown ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
      "This textual convention is an extension of the TruthValue
      convention.  The latter defines a boolean value with possible
      values of true(1) and false(2).  This extension permits the
      additional value of unknown(0), which can be returned as the
      result of a GET operation when an exact true or false value
      of the object cannot be determined."
    SYNTAX       INTEGER { unknown(0), true(1), false(2) }

 -- Port Notifications Group

  efmCuPortNotifications OBJECT IDENTIFIER ::= { efmCuPort 0 }

  efmCuLowRateCrossing NOTIFICATION-TYPE
    OBJECTS {
      ifSpeed,
      efmCuThreshLowRate
    }
    STATUS      current
    DESCRIPTION
      "This notification indicates that the EFMCu port's data rate
      has reached/dropped below or exceeded the low rate threshold,
      specified by efmCuThreshLowRate.

      This notification MAY be sent for the -O subtype ports
      (2BaseTL-O/10PassTS-O) while the port is Up, on the crossing
      event in both directions: from normal (rate is above the
      threshold) to low (rate equals the threshold or below it) and
```

```
          from low to normal.  This notification is not applicable to
          the -R subtypes.

          It is RECOMMENDED that a small debouncing period of 2.5 sec,
          between the detection of the condition and the notification,
          is implemented to prevent simultaneous LinkUp/LinkDown and
          efmCuLowRateCrossing notifications to be sent.

          The adaptive nature of the EFMCu technology allows the port to
          adapt itself to the changes in the copper environment, e.g.,
          an impulse noise, alien crosstalk, or a micro-interruption may
          temporarily drop one or more PMEs in the aggregation group,
          causing a rate degradation of the aggregated EFMCu link.
          The dropped PMEs would then try to re-initialize, possibly at
          a lower rate than before, adjusting the rate to provide
          required target SNR margin.

          Generation of this notification is controlled by the
          efmCuLowRateCrossingEnable object."
        ::= { efmCuPortNotifications 1 }

      -- PCS Port group

      efmCuPortConfTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPortConfEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Table for Configuration of EFMCu 2BASE-TL/10PASS-TS (PCS)
          Ports.  Entries in this table MUST be maintained in a
          persistent manner."
        ::= { efmCuPort 1 }

      efmCuPortConfEntry OBJECT-TYPE
        SYNTAX      EfmCuPortConfEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "An entry in the EFMCu Port Configuration table.
          Each entry represents an EFMCu port indexed by the ifIndex.
          Note that an EFMCu PCS port runs on top of a single
          or multiple PME port(s), which are also indexed by ifIndex."
        INDEX  { ifIndex }
        ::= { efmCuPortConfTable 1 }

      EfmCuPortConfEntry ::=
        SEQUENCE {
          efmCuPAFAdminState               INTEGER,
          efmCuPAFDiscoveryCode            PhysAddress,
          efmCuAdminProfile                EfmProfileIndexList,
          efmCuTargetDataRate              Unsigned32,
          efmCuTargetSnrMgn                Unsigned32,
          efmCuAdaptiveSpectra             TruthValue,
          efmCuThreshLowRate               Unsigned32,
          efmCuLowRateCrossingEnable       TruthValue
        }

      efmCuPAFAdminState  OBJECT-TYPE
        SYNTAX      INTEGER {
          enabled(1),
          disabled(2)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Administrative (desired) state of the PAF of the EFMCu port
          (PCS).
          When 'disabled', PME aggregation will not be performed by the
          PCS.  No more than a single PME can be assigned to this PCS in
          this case.
          When 'enabled', PAF will be performed by the PCS when the link
          is Up, even on a single attached PME, if PAF is supported.

          PCS ports incapable of supporting PAF SHALL return a value of
          'disabled'.  Attempts to 'enable' such ports SHALL be
          rejected.

          A PAF 'enabled' port with multiple PMEs assigned cannot be
          'disabled'.  Attempts to 'disable' such port SHALL be
          rejected, until at most one PME is left assigned.

          Changing PAFAdminState is a traffic-disruptive operation and
          as such SHALL be done when the link is Down.  Attempts to
          change this object SHALL be rejected if the link is Up or
          Initializing.

          This object maps to the Clause 30 attribute aPAFAdminState.

          If a Clause 45 MDIO Interface to the PCS is present, then this
          object maps to the PAF enable bit in the 10P/2B PCS control
          register.

          This object MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] 61.2.2, 45.2.3.18.3"
        ::= { efmCuPortConfEntry 1 }
      efmCuPAFDiscoveryCode  OBJECT-TYPE
        SYNTAX      PhysAddress (SIZE(0|6))
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "PAF Discovery Code of the EFMCu port (PCS).
          A unique 6-octet code used by the Discovery function,
          when PAF is supported.
          PCS ports incapable of supporting PAF SHALL return a
          zero-length octet string on an attempt to read this object.
          An attempt to write to this object SHALL be rejected for such
          ports.
          This object MUST be instantiated for the -O subtype PCS before
          writing operations on the efmCuPAFRemoteDiscoveryCode
          (Set_if_Clear and Clear_if_Same) are performed by PMEs
          associated with the PCS.
          The initial value of this object for -R subtype ports after
          reset is all zeroes.  For -R subtype ports, the value of this
          object cannot be changed directly.  This value may be changed
          as a result of writing operation on the
          efmCuPAFRemoteDiscoveryCode object of remote PME of -O
          subtype, connected to one of the local PMEs associated with
          the PCS.

          Discovery MUST be performed when the link is Down.
          Attempts to change this object MUST be rejected (in case of
          SNMP with the error inconsistentValue), if the link is Up or
          Initializing.

          The PAF Discovery Code maps to the local Discovery code
          variable in PAF (note that it does not have a corresponding
          Clause 45 register)."
        REFERENCE
          "[802.3ah] 61.2.2.8.3, 61.2.2.8.4, 45.2.6.6.1, 45.2.6.8,
          61A.2"
        ::= { efmCuPortConfEntry 2 }

      efmCuAdminProfile  OBJECT-TYPE
        SYNTAX      EfmProfileIndexList
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Desired configuration profile(s), common for all PMEs in the
          EFMCu port.  This object is a list of pointers to entries in
          either efmCuPme2BProfileTable or
          efmCuPme10PProfileTable, depending on the current
          operating SubType of the EFMCu port as indicated by
          efmCuPortSide.
          The value of this object is a list of up to 6 indices of
          profiles.  If this list consists of a single profile index,
          then all PMEs assigned to this EFMCu port SHALL be configured
          according to the profile referenced by that index, unless it
          is overwritten by a corresponding non-zero
          efmCuPmeAdminProfile instance, which takes precedence over
          efmCuAdminProfile.
          A list consisting of more than one index allows each PME
          in the port to be configured according to any profile
          specified in the list.
          By default, this object has a value of 0x01, referencing the
          1st entry in efmCuPme2BProfileTable or
          efmCuPme10PProfileTable.

          This object is writable and readable for the -O subtype
          (2BaseTL-O or 10PassTS-O) EFMCu ports.  It is irrelevant for
          the -R  subtype (2BaseTL-R or 10PassTS-R) ports -- a
          zero-length octet string SHALL be returned on an attempt to
          read this object and an attempt to change this object MUST be
          rejected in this case.

          Note that the current operational profile value is available
          via the efmCuPmeOperProfile object.

          Any modification of this object MUST be performed when the
          link is Down.  Attempts to change this object MUST be
          rejected, if the link is Up or Initializing.
          Attempts to set this object to a list with a member value that
          is not the value of the index for an active entry in the
          corresponding profile table MUST be rejected.

          This object maps to the Clause 30 attribute aProfileSelect.

          This object MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] 30.11.2.1.6"
        DEFVAL { '01'H }
        ::= { efmCuPortConfEntry 3 }

      efmCuTargetDataRate  OBJECT-TYPE
        SYNTAX      Unsigned32(1..100000|999999)
        UNITS       "Kbps"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Desired EFMCu port 'net' (as seen across MII) Data Rate in
          Kbps, to be achieved during initialization, under spectral
          restrictions placed on each PME via efmCuAdminProfile or
          efmCuPmeAdminProfile, with the desired SNR margin specified by
          efmCuTargetSnrMgn.
          In case of PAF, this object represents a sum of individual PME
          data rates, modified to compensate for fragmentation and
          64/65-octet encapsulation overhead (e.g., target data rate of
          10 Mbps SHALL allow lossless transmission of a full-duplex
          10 Mbps Ethernet frame stream with minimal inter-frame gap).

          The value is limited above by 100 Mbps as this is the max
          burst rate across MII for EFMCu ports.

          The value between 1 and 100000 indicates that the total data
          rate (ifSpeed) of the EFMCu port after initialization SHALL be
          equal to the target data rate or less, if the target data rate
          cannot be achieved under spectral restrictions specified by
          efmCuAdminProfile/efmCuPmeAdminProfile and with the desired
          SNR margin.  In case the copper environment allows a higher
          total data rate to be achieved than that specified by the
          target, the excess capability SHALL be either converted to
          additional SNR margin or reclaimed by minimizing transmit
          power as controlled by efmCuAdaptiveSpectra.

          The value of 999999 means that the target data rate is not
          fixed and SHALL be set to the maximum attainable rate during
          initialization (Best Effort), under specified spectral
          restrictions and with the desired SNR margin.

          This object is read-write for the -O subtype EFMCu ports
          (2BaseTL-O/10PassTS-O) and not available for the -R subtypes.

          Changing of the Target Data Rate MUST be performed when the
          link is Down.  Attempts to change this object MUST be rejected
          (in case of SNMP with the error inconsistentValue), if the
          link is Up or Initializing.

          Note that the current Data Rate of the EFMCu port is
          represented by the ifSpeed object of IF-MIB.

          This object MUST be maintained in a persistent manner."
        ::= { efmCuPortConfEntry 4 }

      efmCuTargetSnrMgn  OBJECT-TYPE
        SYNTAX      Unsigned32(0..21)
        UNITS       "dB"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Desired EFMCu port SNR margin to be achieved on all PMEs
          assigned to the port, during initialization. (The SNR margin
          is the difference between the desired SNR and the actual SNR).

          Note that 802.3ah recommends using a default target SNR margin
          of 5 dB for 2BASE-TL ports and 6 dB for 10PASS-TS ports in
          order to achieve a mean Bit Error Rate (BER) of 10^-7 at the
          PMA service interface.

          This object is read-write for the -O subtype EFMCu ports
          (2BaseTL-O/10PassTS-O) and not available for the -R subtypes.

          Changing of the target SNR margin MUST be performed when the
          link is Down.  Attempts to change this object MUST be rejected
          (in case of SNMP with the error inconsistentValue), if the
          link is Up or Initializing.

          Note that the current SNR margin of the PMEs comprising the
          EFMCu port is represented by efmCuPmeSnrMgn.

          This object MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] 61.1.2"
        ::= { efmCuPortConfEntry 5 }

      efmCuAdaptiveSpectra  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Indicates how to utilize excess capacity when the copper
          environment allows a higher total data rate to be achieved
          than that specified by the efmCuTargetDataRate.

          A value of true(1) indicates that the excess capability SHALL
          be reclaimed by minimizing transmit power, e.g., using higher
          constellations and Power Back-Off, in order to reduce
          interference to other copper pairs in the binder and the
          adverse impact to link/system performance.

          A value of false(2) indicates that the excess capability SHALL
          be converted to additional SNR margin and spread evenly across
          all active PMEs assigned to the (PCS) port, to increase link
          robustness.

          This object is read-write for the -O subtype EFMCu ports
          (2BaseTL-O/10PassTS-O) and not available for the -R subtypes.

          Changing of this object MUST be performed when the link is
          Down.  Attempts to change this object MUST be rejected (in
          case of SNMP with the error inconsistentValue), if the link
          is Up or Initializing.

          This object MUST be maintained in a persistent manner."
        ::= { efmCuPortConfEntry 6 }

      efmCuThreshLowRate  OBJECT-TYPE
        SYNTAX      Unsigned32(1..100000)
        UNITS       "Kbps"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "This object configures the EFMCu port low-rate crossing alarm
          threshold.  When the current value of ifSpeed for this port
          reaches/drops below or exceeds this threshold, an
          efmCuLowRateCrossing notification MAY be generated if enabled
          by efmCuLowRateCrossingEnable.

          This object is read-write for the -O subtype EFMCu ports
          (2BaseTL-O/10PassTS-O) and not available for the -R subtypes.

          This object MUST be maintained in a persistent manner."
        ::= { efmCuPortConfEntry 7 }

      efmCuLowRateCrossingEnable  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Indicates whether efmCuLowRateCrossing notifications should
          be generated for this interface.

          A value of true(1) indicates that efmCuLowRateCrossing
          notification is enabled.  A value of false(2) indicates that
          the notification is disabled.

          This object is read-write for the -O subtype EFMCu ports
          (2BaseTL-O/10PassTS-O) and not available for the -R subtypes.

          This object MUST be maintained in a persistent manner."
        ::= { efmCuPortConfEntry 8 }


      efmCuPortCapabilityTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPortCapabilityEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Table for Capabilities of EFMCu 2BASE-TL/10PASS-TS (PCS)
          Ports.  Entries in this table MUST be maintained in a
          persistent manner"
        ::= { efmCuPort 2 }

      efmCuPortCapabilityEntry OBJECT-TYPE
        SYNTAX      EfmCuPortCapabilityEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "An entry in the EFMCu Port Capability table.
          Each entry represents an EFMCu port indexed by the ifIndex.
          Note that an EFMCu PCS port runs on top of a single
          or multiple PME port(s), which are also indexed by ifIndex."
        INDEX  { ifIndex }
        ::= { efmCuPortCapabilityTable 1 }

      EfmCuPortCapabilityEntry ::=
        SEQUENCE {
          efmCuPAFSupported                TruthValue,
          efmCuPeerPAFSupported            EfmTruthValueOrUnknown,
          efmCuPAFCapacity                 Unsigned32,
          efmCuPeerPAFCapacity             Unsigned32
        }

      efmCuPAFSupported  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "PME Aggregation Function (PAF) capability of the EFMCu port
          (PCS).
          This object has a value of true(1) when the PCS can perform
          PME aggregation on the available PMEs.
          Ports incapable of PAF SHALL return a value of false(2).

          This object maps to the Clause 30 attribute aPAFSupported.

          If a Clause 45 MDIO Interface to the PCS is present,
          then this object maps to the PAF available bit in the
          10P/2B capability register."
        REFERENCE
          "[802.3ah] 61.2.2, 30.11.1.1.4, 45.2.3.17.1"
        ::= { efmCuPortCapabilityEntry 1 }

      efmCuPeerPAFSupported  OBJECT-TYPE
        SYNTAX      EfmTruthValueOrUnknown
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "PME Aggregation Function (PAF) capability of the EFMCu port
          (PCS) link partner.
          This object has a value of true(1) when the remote PCS can
          perform PME aggregation on its available PMEs.
          Ports whose peers are incapable of PAF SHALL return a value
          of false(2).
          Ports whose peers cannot be reached because of the link
          state SHALL return a value of unknown(0).

          This object maps to the Clause 30 attribute
          aRemotePAFSupported.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the Remote PAF supported bit in the
          10P/2B capability register."
        REFERENCE
          "[802.3ah] 61.2.2, 30.11.1.1.9, 45.2.3.17.2"
        ::= { efmCuPortCapabilityEntry 2 }

      efmCuPAFCapacity  OBJECT-TYPE
        SYNTAX      Unsigned32 (1..32)
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "Number of PMEs that can be aggregated by the local PAF.
          The number of PMEs currently assigned to a particular
          EFMCu port (efmCuNumPMEs) is never greater than
          efmCuPAFCapacity.

          This object maps to the Clause 30 attribute
          aLocalPAFCapacity."
        REFERENCE
          "[802.3ah] 61.2.2, 30.11.1.1.6"
        ::= { efmCuPortCapabilityEntry 3 }

      efmCuPeerPAFCapacity  OBJECT-TYPE
        SYNTAX      Unsigned32 (0|1..32)
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "Number of PMEs that can be aggregated by the PAF of the peer
          PHY (PCS port).
          A value of 0 is returned when peer PAF capacity is unknown
          (peer cannot be reached).
```

```
           This object maps to the Clause 30 attribute
           aRemotePAFCapacity."
         REFERENCE
           "[802.3ah] 61.2.2, 30.11.1.1.10"
         ::= { efmCuPortCapabilityEntry 4 }

       efmCuPortStatusTable OBJECT-TYPE
         SYNTAX      SEQUENCE OF EfmCuPortStatusEntry
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
           "This table provides overall status information of EFMCu
           2BASE-TL/10PASS-TS ports, complementing the generic status
           information from the ifTable of IF-MIB and ifMauTable of
           MAU-MIB.  Additional status information about connected PMEs
           is available from the efmCuPmeStatusTable.

           This table contains live data from the equipment.  As such,
           it is NOT persistent."
         ::= { efmCuPort 3 }

       efmCuPortStatusEntry OBJECT-TYPE
         SYNTAX      EfmCuPortStatusEntry
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
           "An entry in the EFMCu Port Status table.
           Each entry represents an EFMCu port indexed by the ifIndex.
           Note that an EFMCu PCS port runs on top of a single
           or multiple PME port(s), which are also indexed by ifIndex."
         INDEX  { ifIndex }
         ::= { efmCuPortStatusTable 1 }

       EfmCuPortStatusEntry ::=
         SEQUENCE {
           efmCuFltStatus                  BITS,
           efmCuPortSide                   INTEGER,
           efmCuNumPMEs                    Unsigned32,
           efmCuPAFInErrors                Counter32,
           efmCuPAFInSmallFragments        Counter32,
           efmCuPAFInLargeFragments        Counter32,
           efmCuPAFInBadFragments          Counter32,
           efmCuPAFInLostFragments         Counter32,
           efmCuPAFInLostStarts            Counter32,
           efmCuPAFInLostEnds              Counter32,
           efmCuPAFInOverflows             Counter32
         }
```

```
      efmCuFltStatus  OBJECT-TYPE
        SYNTAX      BITS {
          noPeer(0),
          peerPowerLoss(1),
          pmeSubTypeMismatch(2),
          lowRate(3)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "EFMCu (PCS) port Fault Status.  This is a bitmap of possible
          conditions.  The various bit positions are:
            noPeer              - the peer PHY cannot be reached (e.g.,
                                  no PMEs attached, all PMEs are Down,
                                  etc.). More info is available in
                                  efmCuPmeFltStatus.
            peerPowerLoss       - the peer PHY has indicated impending
                                  unit failure due to loss of local
                                  power ('Dying Gasp').
            pmeSubTypeMismatch  - local PMEs in the aggregation group
                                  are not of the same subtype, e.g.,
                                  some PMEs in the local device are -O
                                  while others are -R subtype.
            lowRate             - ifSpeed of the port reached or dropped
                                  below efmCuThreshLowRate.

          This object is intended to supplement the ifOperStatus object
          in IF-MIB and ifMauMediaAvailable in MAU-MIB.

          Additional information is available via the efmCuPmeFltStatus
          object for each PME in the aggregation group (single PME if
          PAF is disabled)."
        REFERENCE
          "IF-MIB, ifOperStatus; MAU-MIB, ifMauMediaAvailable;
           efmCuPmeFltStatus"
        ::= { efmCuPortStatusEntry 1 }

      efmCuPortSide  OBJECT-TYPE
        SYNTAX      INTEGER {
          subscriber(1),
          office(2),
          unknown(3)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "EFM port mode of operation (subtype).
          The value of 'subscriber' indicates that the port is
          designated as '-R' subtype (all PMEs assigned to this port are
          of subtype '-R').
          The value of the 'office' indicates that the port is
          designated as '-O' subtype (all PMEs assigned to this port are
          of subtype '-O').
          The value of 'unknown' indicates that the port has no assigned
          PMEs yet or that the assigned PMEs are not of the same side
          (subTypePMEMismatch).

          This object partially maps to the Clause 30 attribute
          aPhyEnd."
        REFERENCE
           "[802.3ah] 61.1, 30.11.1.1.2"
        ::= { efmCuPortStatusEntry 2 }

      efmCuNumPMEs  OBJECT-TYPE
        SYNTAX      Unsigned32 (0..32)
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of PMEs that is currently aggregated by the local
          PAF (assigned to the EFMCu port using the ifStackTable).
          This number is never greater than efmCuPAFCapacity.

          This object SHALL be automatically incremented or decremented
          when a PME is added or deleted to/from the EFMCu port using
          the ifStackTable."
        REFERENCE
          "[802.3ah] 61.2.2, 30.11.1.1.6"
        ::= { efmCuPortStatusEntry 3 }

      efmCuPAFInErrors OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of fragments that have been received across the
          gamma interface with RxErr asserted and discarded.
          This read-only counter is inactive (not incremented) when the
          PAF is unsupported or disabled.  Upon disabling the PAF, the
          counter retains its previous value.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the 10P/2B PAF RX error register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.3.21"
        ::= { efmCuPortStatusEntry 4 }

      efmCuPAFInSmallFragments OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of fragments smaller than minFragmentSize
          (64 bytes) that have been received across the gamma interface
          and discarded.
          This read-only counter is inactive when the PAF is
          unsupported or disabled.  Upon disabling the PAF, the counter
          retains its previous value.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the 10P/2B PAF small fragments register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.3.22"
        ::= { efmCuPortStatusEntry 5 }

      efmCuPAFInLargeFragments OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of fragments larger than maxFragmentSize
          (512 bytes) that have been received across the gamma interface
          and discarded.
          This read-only counter is inactive when the PAF is
          unsupported or disabled.  Upon disabling the PAF, the counter
          retains its previous value.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the 10P/2B PAF large fragments register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
```

```
         "[802.3ah] 45.2.3.23"
       ::= { efmCuPortStatusEntry 6 }

     efmCuPAFInBadFragments OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
         "The number of fragments that do not fit into the sequence
         expected by the frame assembly function and that have been
         received across the gamma interface and discarded (the
         frame buffer is flushed to the next valid frame start).
         This read-only counter is inactive when the PAF is
         unsupported or disabled.  Upon disabling the PAF, the counter
         retains its previous value.

         If a Clause 45 MDIO Interface to the PCS is present, then
         this object maps to the 10P/2B PAF bad fragments register.

         Discontinuities in the value of this counter can occur at
         re-initialization of the management system, and at other times
         as indicated by the value of ifCounterDiscontinuityTime,
         defined in IF-MIB."
       REFERENCE
         "[802.3ah] 45.2.3.25"
       ::= { efmCuPortStatusEntry 7 }

     efmCuPAFInLostFragments OBJECT-TYPE
       SYNTAX      Counter32
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
         "The number of gaps in the sequence of fragments that have
         been received across the gamma interface (the frame buffer is
         flushed to the next valid frame start, when fragment/fragments
         expected by the frame assembly function is/are not received).
         This read-only counter is inactive when the PAF is
         unsupported or disabled.  Upon disabling the PAF, the counter
         retains its previous value.

         If a Clause 45 MDIO Interface to the PCS is present, then
         this object maps to the 10P/2B PAF lost fragment register.

         Discontinuities in the value of this counter can occur at
         re-initialization of the management system, and at other times
         as indicated by the value of ifCounterDiscontinuityTime,
         defined in IF-MIB."
       REFERENCE
```

```
          "[802.3ah] 45.2.3.26"
        ::= { efmCuPortStatusEntry 8 }

      efmCuPAFInLostStarts OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of missing StartOfPacket indicators expected by
          the frame assembly function.
          This read-only counter is inactive when the PAF is
          unsupported or disabled.  Upon disabling the PAF, the counter
          retains its previous value.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the 10P/2B PAF lost start of fragment
          register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.3.27"
        ::= { efmCuPortStatusEntry 9 }

      efmCuPAFInLostEnds OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of missing EndOfPacket indicators expected by the
          frame assembly function.
          This read-only counter is inactive when the PAF is
          unsupported or disabled.  Upon disabling the PAF, the counter
          retains its previous value.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the 10P/2B PAF lost start of fragment
          register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.3.28"
        ::= { efmCuPortStatusEntry 10 }
      efmCuPAFInOverflows OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of fragments, received across the gamma interface
          and discarded, which would have caused the frame assembly
          buffer to overflow.
          This read-only counter is inactive when the PAF is
          unsupported or disabled.  Upon disabling the PAF, the counter
          retains its previous value.

          If a Clause 45 MDIO Interface to the PCS is present, then
          this object maps to the 10P/2B PAF overflow register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.3.24"
        ::= { efmCuPortStatusEntry 11 }

     -- PME Notifications Group

      efmCuPmeNotifications OBJECT IDENTIFIER ::= { efmCuPme 0 }

      efmCuPmeLineAtnCrossing NOTIFICATION-TYPE
        OBJECTS {
          efmCuPmeLineAtn,
          efmCuPmeThreshLineAtn
        }
        STATUS      current
        DESCRIPTION
          "This notification indicates that the loop attenuation
          threshold (as per the efmCuPmeThreshLineAtn
          value) has been reached/exceeded for the 2BASE-TL/10PASS-TS
          PME.  This notification MAY be sent on the crossing event in
          both directions: from normal to exceeded and from exceeded
          to normal.

          It is RECOMMENDED that a small debouncing period of 2.5 sec,
          between the detection of the condition and the notification,
          is implemented to prevent intermittent notifications from
          being sent.

          Generation of this notification is controlled by the
          efmCuPmeLineAtnCrossingEnable object."
        ::= { efmCuPmeNotifications 1 }

      efmCuPmeSnrMgnCrossing NOTIFICATION-TYPE
        OBJECTS {
          efmCuPmeSnrMgn,
          efmCuPmeThreshSnrMgn
        }
        STATUS      current
        DESCRIPTION
          "This notification indicates that the SNR margin threshold
          (as per the efmCuPmeThreshSnrMgn value) has been
          reached/exceeded for the 2BASE-TL/10PASS-TS PME.
          This notification MAY be sent on the crossing event in
          both directions: from normal to exceeded and from exceeded
          to normal.

          It is RECOMMENDED that a small debouncing period of 2.5 sec,
          between the detection of the condition and the notification,
          is implemented to prevent intermittent notifications from
          being sent.

          Generation of this notification is controlled by the
          efmCuPmeSnrMgnCrossingEnable object."
        ::= { efmCuPmeNotifications 2 }

      efmCuPmeDeviceFault NOTIFICATION-TYPE
        OBJECTS {
          efmCuPmeFltStatus
        }
        STATUS      current
        DESCRIPTION
          "This notification indicates that a fault in the PME has been
          detected by a vendor-specific diagnostic or a self-test.

          Generation of this notification is controlled by the
          efmCuPmeDeviceFaultEnable object."
        ::= { efmCuPmeNotifications 3 }

      efmCuPmeConfigInitFailure NOTIFICATION-TYPE
        OBJECTS {
          efmCuPmeFltStatus,
          efmCuAdminProfile,
          efmCuPmeAdminProfile
        }
        STATUS      current
        DESCRIPTION
          "This notification indicates that PME initialization has
          failed, due to inability of the PME link to achieve the
          requested configuration profile.

          Generation of this notification is controlled by the
          efmCuPmeConfigInitFailEnable object."
        ::= { efmCuPmeNotifications 4 }

      efmCuPmeProtocolInitFailure NOTIFICATION-TYPE
        OBJECTS {
          efmCuPmeFltStatus,
          efmCuPmeOperSubType
        }
        STATUS     current
        DESCRIPTION
          "This notification indicates that the peer PME was using
          an incompatible protocol during initialization.

          Generation of this notification is controlled by the
          efmCuPmeProtocolInitFailEnable object."
        ::= { efmCuPmeNotifications 5 }

      -- The PME group

      efmCuPmeConfTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPmeConfEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Table for Configuration of common aspects for EFMCu
          2BASE-TL/10PASS-TS PME ports (modems).  Configuration of
          aspects specific to 2BASE-TL or 10PASS-TS PME types is
          represented in efmCuPme2BConfTable and efmCuPme10PConfTable,
          respectively.

          Entries in this table MUST be maintained in a persistent
          manner."
        ::= { efmCuPme 1 }

      efmCuPmeConfEntry OBJECT-TYPE
        SYNTAX      EfmCuPmeConfEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "An entry in the EFMCu PME Configuration table.
          Each entry represents common aspects of an EFMCu PME port
          indexed by the ifIndex.  Note that an EFMCu PME port can be
          stacked below a single PCS port, also indexed by ifIndex,
          possibly together with other PME ports if PAF is enabled."
        INDEX  { ifIndex }
        ::= { efmCuPmeConfTable 1 }

      EfmCuPmeConfEntry ::=
        SEQUENCE {
          efmCuPmeAdminSubType           INTEGER,
          efmCuPmeAdminProfile           EfmProfileIndexOrZero,
          efmCuPAFRemoteDiscoveryCode    PhysAddress,
          efmCuPmeThreshLineAtn          Integer32,
          efmCuPmeThreshSnrMgn           Integer32,
          efmCuPmeLineAtnCrossingEnable  TruthValue,
          efmCuPmeSnrMgnCrossingEnable   TruthValue,
          efmCuPmeDeviceFaultEnable      TruthValue,
          efmCuPmeConfigInitFailEnable   TruthValue,
          efmCuPmeProtocolInitFailEnable TruthValue
        }

      efmCuPmeAdminSubType  OBJECT-TYPE
        SYNTAX      INTEGER {
          ieee2BaseTLO(1),
          ieee2BaseTLR(2),
          ieee10PassTSO(3),
          ieee10PassTSR(4),
          ieee2BaseTLor10PassTSR(5),
          ieee2BaseTLor10PassTSO(6),
          ieee10PassTSor2BaseTLO(7)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Administrative (desired) subtype of the PME.
          Possible values are:
            ieee2BaseTLO           - PME SHALL operate as 2BaseTL-O
            ieee2BaseTLR           - PME SHALL operate as 2BaseTL-R
            ieee10PassTSO          - PME SHALL operate as 10PassTS-O
            ieee10PassTSR          - PME SHALL operate as 10PassTS-R
            ieee2BaseTLor10PassTSR - PME SHALL operate as 2BaseTL-R or
                                     10PassTS-R.  The actual value will
                                     be set by the -O link partner
                                     during initialization (handshake).
            ieee2BaseTLor10PassTSO - PME SHALL operate as 2BaseTL-O
                                     (preferred) or 10PassTS-O.  The
                                     actual value will be set during
                                     initialization depending on the -R
                                     link partner capability (i.e., if
                                     -R is incapable of the preferred
                                     2BaseTL mode, 10PassTS will be
                                     used).
            ieee10PassTSor2BaseTLO - PME SHALL operate as 10PassTS-O
```

```
                                     (preferred) or 2BaseTL-O.  The
                                     actual value will be set during
                                     initialization depending on the -R
                                     link partner capability (i.e., if
                                     -R is incapable of the preferred
                                     10PassTS mode, 2BaseTL will be
                                     used).

          Changing efmCuPmeAdminSubType is a traffic-disruptive
          operation and as such SHALL be done when the link is Down.
          Attempts to change this object SHALL be rejected if the link
          is Up or Initializing.
          Attempts to change this object to an unsupported subtype
          (see efmCuPmeSubTypesSupported) SHALL be rejected.

          The current operational subtype is indicated by the
          efmCuPmeOperSubType variable.

          If a Clause 45 MDIO Interface to the PMA/PMD is present, then
          this object combines values of the Port subtype select bits
          and the PMA/PMD type selection bits in the 10P/2B PMA/PMD
          control register."
        REFERENCE
          "[802.3ah] 61.1, 45.2.1.11.4, 45.2.1.11.7"
        ::= { efmCuPmeConfEntry 1 }

      efmCuPmeAdminProfile  OBJECT-TYPE
        SYNTAX      EfmProfileIndexOrZero
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Desired PME configuration profile.  This object is a pointer
          to an entry in either the efmCuPme2BProfileTable or the
          efmCuPme10PProfileTable, depending on the current operating
          SubType of the PME.  The value of this object is the index of
          the referenced profile.
          The value of zero (default) indicates that the PME is
          configured via the efmCuAdminProfile object for the PCS port
          to which this PME is assigned.  That is, the profile
          referenced by efmCuPmeAdminProfile takes precedence
          over the profile(s) referenced by efmCuAdminProfile.

          This object is writable and readable for the CO subtype PMEs
          (2BaseTL-O or 10PassTS-O). It is irrelevant for the CPE
          subtype (2BaseTL-R or 10PassTS-R) -- a zero value SHALL be
          returned on an attempt to read this object and any attempt
          to change this object MUST be rejected in this case.
          Note that the current operational profile value is available
          via efmCuPmeOperProfile object.

          Any modification of this object MUST be performed when the
          link is Down.  Attempts to change this object MUST be
          rejected, if the link is Up or Initializing.

          Attempts to set this object to a value that is not the value
          of the index for an active entry in the corresponding profile
          table MUST be rejected.

          This object maps to the Clause 30 attribute aProfileSelect.

          This object MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] 30.11.2.1.6"
        DEFVAL { 0 }
        ::= { efmCuPmeConfEntry 2 }

      efmCuPAFRemoteDiscoveryCode  OBJECT-TYPE
        SYNTAX      PhysAddress (SIZE(0|6))
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "PAF Remote Discovery Code of the PME port at the CO.
          The 6-octet Discovery Code of the peer PCS connected via
          the PME.
          Reading this object results in a Discovery Get operation.
          Setting this object to all zeroes results in a Discovery
          Clear_if_Same operation (the value of efmCuPAFDiscoveryCode
          at the peer PCS SHALL be the same as efmCuPAFDiscoveryCode of
          the local PCS associated with the PME for the operation to
          succeed).
          Writing a non-zero value to this object results in a
          Discovery Set_if_Clear operation.
          A zero-length octet string SHALL be returned on an attempt to
          read this object when PAF aggregation is not enabled.

          This object is irrelevant in CPE port (-R) subtypes: in this
          case, a zero-length octet string SHALL be returned on an
          attempt to read this object; writing to this object SHALL
          be rejected.

          Discovery MUST be performed when the link is Down.
          Attempts to change this object MUST be rejected (in case of
          SNMP with the error inconsistentValue), if the link is Up or
          Initializing.
          If a Clause 45 MDIO Interface to the PMA/PMD is present, then
          this object is a function of 10P/2B aggregation discovery
          control register, Discovery operation result bits in 10P/2B
          aggregation and discovery status register and
          10P/2B aggregation discovery code register."
        REFERENCE
          "[802.3ah] 61.2.2.8.4, 45.2.6.6-45.2.6.8"
        ::= { efmCuPmeConfEntry 3 }

      efmCuPmeThreshLineAtn  OBJECT-TYPE
        SYNTAX  Integer32(-127..128)
        UNITS       "dB"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Desired Line Attenuation threshold for the 2B/10P PME.
          This object configures the line attenuation alarm threshold.
          When the current value of Line Attenuation reaches or
          exceeds this threshold, an efmCuPmeLineAtnCrossing
          notification MAY be generated, if enabled by
          efmCuPmeLineAtnCrossingEnable.

          This object is writable for the CO subtype PMEs (-O).
          It is read-only for the CPE subtype (-R).

          Changing of the Line Attenuation threshold MUST be performed
          when the link is Down.  Attempts to change this object MUST be
          rejected (in case of SNMP with the error inconsistentValue),
          if the link is Up or Initializing.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object maps to the loop attenuation threshold bits in
          the 2B PMD line quality thresholds register."
        REFERENCE
          "[802.3ah] 45.2.1.36"
        ::= { efmCuPmeConfEntry 4 }

      efmCuPmeThreshSnrMgn  OBJECT-TYPE
        SYNTAX      Integer32(-127..128)
        UNITS       "dB"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Desired SNR margin threshold for the 2B/10P PME.
          This object configures the SNR margin alarm threshold.
          When the current value of SNR margin reaches or exceeds this
          threshold, an efmCuPmeSnrMgnCrossing notification MAY be
          generated, if enabled by efmCuPmeSnrMgnCrossingEnable.
          This object is writable for the CO subtype PMEs
          (2BaseTL-O/10PassTS-O).  It is read-only for the CPE subtype
          (2BaseTL-R/10PassTS-R).

          Changing of the SNR margin threshold MUST be performed when
          the link is Down.  Attempts to change this object MUST be
          rejected (in case of SNMP with the error inconsistentValue),
          if the link is Up or Initializing.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object maps to the SNR margin threshold bits in the 2B PMD
          line quality thresholds register."
        REFERENCE
          "[802.3ah] 45.2.1.36"
        ::= { efmCuPmeConfEntry 5 }

      efmCuPmeLineAtnCrossingEnable  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Indicates whether efmCuPmeLineAtnCrossing notifications
          should be generated for this interface.

          A value of true(1) indicates that efmCuPmeLineAtnCrossing
          notification is enabled.  A value of false(2) indicates that
          the notification is disabled."
        ::= { efmCuPmeConfEntry 6 }

      efmCuPmeSnrMgnCrossingEnable  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Indicates whether efmCuPmeSnrMgnCrossing notifications
          should be generated for this interface.

          A value of true(1) indicates that efmCuPmeSnrMgnCrossing
          notification is enabled.  A value of false(2) indicates that
          the notification is disabled."
        ::= { efmCuPmeConfEntry 7 }

      efmCuPmeDeviceFaultEnable  OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "Indicates whether efmCuPmeDeviceFault notifications
```

```
           should be generated for this interface.

           A value of true(1) indicates that efmCuPmeDeviceFault
           notification is enabled.  A value of false(2) indicates that
           the notification is disabled."
         ::= { efmCuPmeConfEntry 8 }

       efmCuPmeConfigInitFailEnable  OBJECT-TYPE
         SYNTAX      TruthValue
         MAX-ACCESS  read-write
         STATUS      current
         DESCRIPTION
           "Indicates whether efmCuPmeConfigInitFailure notifications
           should be generated for this interface.

           A value of true(1) indicates that efmCuPmeConfigInitFailure
           notification is enabled.  A value of false(2) indicates that
           the notification is disabled."
         ::= { efmCuPmeConfEntry 9 }

       efmCuPmeProtocolInitFailEnable  OBJECT-TYPE
         SYNTAX      TruthValue
         MAX-ACCESS  read-write
         STATUS      current
         DESCRIPTION
           "Indicates whether efmCuPmeProtocolInitFailure notifications
           should be generated for this interface.

           A value of true(1) indicates that efmCuPmeProtocolInitFailure
           notification is enabled.  A value of false(2) indicates that
           the notification is disabled."
         ::= { efmCuPmeConfEntry 10 }


       efmCuPmeCapabilityTable OBJECT-TYPE
         SYNTAX      SEQUENCE OF EfmCuPmeCapabilityEntry
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
           "Table for the configuration of common aspects for EFMCu
           2BASE-TL/10PASS-TS PME ports (modems).  The configuration of
           aspects specific to 2BASE-TL or 10PASS-TS PME types is
           represented in the efmCuPme2BConfTable and the
           efmCuPme10PConfTable, respectively.

           Entries in this table MUST be maintained in a persistent
           manner."
         ::= { efmCuPme 2 }
```

```
      efmCuPmeCapabilityEntry OBJECT-TYPE
        SYNTAX      EfmCuPmeCapabilityEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "An entry in the EFMCu PME Capability table.
          Each entry represents common aspects of an EFMCu PME port
          indexed by the ifIndex.  Note that an EFMCu PME port can be
          stacked below a single PCS port, also indexed by ifIndex,
          possibly together with other PME ports if PAF is enabled."
        INDEX  { ifIndex }
        ::= { efmCuPmeCapabilityTable 1 }

      EfmCuPmeCapabilityEntry ::=
        SEQUENCE {
          efmCuPmeSubTypesSupported     BITS
        }

      efmCuPmeSubTypesSupported  OBJECT-TYPE
        SYNTAX      BITS {
          ieee2BaseTLO(0),
          ieee2BaseTLR(1),
          ieee10PassTSO(2),
          ieee10PassTSR(3)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "PME supported subtypes.  This is a bitmap of possible
          subtypes.  The various bit positions are:
            ieee2BaseTLO    - PME is capable of operating as 2BaseTL-O
            ieee2BaseTLR    - PME is capable of operating as 2BaseTL-R
            ieee10PassTSO   - PME is capable of operating as 10PassTS-O
            ieee10PassTSR   - PME is capable of operating as 10PassTS-R

          The desired mode of operation is determined by
          efmCuPmeAdminSubType, while efmCuPmeOperSubType reflects the
          current operating mode.

          If a Clause 45 MDIO Interface to the PCS is present, then this
          object combines the 10PASS-TS capable and 2BASE-TL capable
          bits in the 10P/2B PMA/PMD speed ability register and the
          CO supported and CPE supported bits in the 10P/2B PMA/PMD
          status register."
        REFERENCE
          "[802.3ah] 61.1, 45.2.1.4.1, 45.2.1.4.2, 45.2.1.12.2,
          45.2.1.12.3"
        ::= { efmCuPmeCapabilityEntry 1 }
```

```
   efmCuPmeStatusTable OBJECT-TYPE
     SYNTAX      SEQUENCE OF EfmCuPmeStatusEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
       "This table provides common status information of EFMCu
       2BASE-TL/10PASS-TS PME ports.  Status information specific
       to 10PASS-TS PME is represented in efmCuPme10PStatusTable.

       This table contains live data from the equipment.  As such,
       it is NOT persistent."
     ::= { efmCuPme 3 }

   efmCuPmeStatusEntry OBJECT-TYPE
     SYNTAX      EfmCuPmeStatusEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
       "An entry in the EFMCu PME Status table.
       Each entry represents common aspects of an EFMCu PME port
       indexed by the ifIndex.  Note that an EFMCu PME port can be
       stacked below a single PCS port, also indexed by ifIndex,
       possibly together with other PME ports if PAF is enabled."
     INDEX  { ifIndex }
     ::= { efmCuPmeStatusTable 1 }

   EfmCuPmeStatusEntry ::=
     SEQUENCE {
       efmCuPmeOperStatus            INTEGER,
       efmCuPmeFltStatus             BITS,
       efmCuPmeOperSubType           INTEGER,
       efmCuPmeOperProfile           EfmProfileIndexOrZero,
       efmCuPmeSnrMgn                Integer32,
       efmCuPmePeerSnrMgn            Integer32,
       efmCuPmeLineAtn               Integer32,
       efmCuPmePeerLineAtn           Integer32,
       efmCuPmeEquivalentLength      Unsigned32,
       efmCuPmeTCCodingErrors        Counter32,
       efmCuPmeTCCrcErrors           Counter32
     }

   efmCuPmeOperStatus  OBJECT-TYPE
     SYNTAX      INTEGER {
       up(1),
       downNotReady(2),
       downReady(3),
       init(4)
     }
```

```
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "Current PME link Operational Status.  Possible values are:
            up(1)           - The link is Up and ready to pass
                              64/65-octet encoded frames or fragments.
            downNotReady(2) - The link is Down and the PME does not
                              detect Handshake tones from its peer.
                              This value may indicate a possible
                              problem with the peer PME.
            downReady(3)    - The link is Down and the PME detects
                              Handshake tones from its peer.
            init(4)         - The link is Initializing, as a result of
                              ifAdminStatus being set to ’up’ for a
                              particular PME or a PCS to which the PME
                              is connected.

          This object is intended to supplement the Down(2) state of
          ifOperStatus.

          This object partially maps to the Clause 30 attribute
          aPMEStatus.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object partially maps to PMA/PMD link status bits in 10P/2B
          PMA/PMD status register."
        REFERENCE
          "[802.3ah] 30.11.2.1.3, 45.2.1.12.4"
        ::= { efmCuPmeStatusEntry 1 }

      efmCuPmeFltStatus  OBJECT-TYPE
        SYNTAX      BITS {
          lossOfFraming(0),
          snrMgnDefect(1),
          lineAtnDefect(2),
          deviceFault(3),
          configInitFailure(4),
          protocolInitFailure(5)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "Current/Last PME link Fault Status.  This is a bitmap of
          possible conditions.  The various bit positions are:

            lossOfFraming       - Loss of Framing for 10P or
                                  Loss of Sync word for 2B PMD or
                                  Loss of 64/65-octet framing.
            snrMgnDefect        - SNR margin dropped below the
                                  threshold.
            lineAtnDefect       - Line Attenuation exceeds the
                                  threshold.
            deviceFault         - Indicates a vendor-dependent
                                  diagnostic or self-test fault
                                  has been detected.
            configInitFailure   - Configuration initialization failure,
                                  due to inability of the PME link to
                                  support the configuration profile,
                                  requested during initialization.
            protocolInitFailure - Protocol initialization failure, due
                                  to an incompatible protocol used by
                                  the peer PME during init (that could
                                  happen if a peer PMD is a regular
                                  G.SDHSL/VDSL modem instead of a
                                  2BASE-TL/10PASS-TS PME).

          This object is intended to supplement ifOperStatus in IF-MIB.

          This object holds information about the last fault.
          efmCuPmeFltStatus is cleared by the device restart.
          In addition, lossOfFraming, configInitFailure, and
          protocolInitFailure are cleared by PME init;
          deviceFault is cleared by successful diagnostics/test;
          snrMgnDefect and lineAtnDefect are cleared by SNR margin
          and Line attenuation, respectively, returning to norm and by
          PME init.

          This object partially maps to the Clause 30 attribute
          aPMEStatus.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object consolidates information from various PMA/PMD
          registers, namely: Fault bit in PMA/PMD status 1 register,
          10P/2B PMA/PMD link loss register,
          10P outgoing indicator bits status register,
          10P incoming indicator bits status register,
          2B state defects register."
        REFERENCE
          "[802.3ah] 30.11.2.1.3, 45.2.1.2.1, 45.2.1.38,
          45.2.1.39, 45.2.1.54"
        ::= { efmCuPmeStatusEntry 2 }

      efmCuPmeOperSubType  OBJECT-TYPE
        SYNTAX      INTEGER {
          ieee2BaseTLO(1),
          ieee2BaseTLR(2),
          ieee10PassTSO(3),
          ieee10PassTSR(4)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "Current operational subtype of the PME.
          Possible values are:
            ieee2BaseTLO           - PME operates as 2BaseTL-O
            ieee2BaseTLR           - PME operates as 2BaseTL-R
            ieee10PassTSO          - PME operates as 10PassTS-O
            ieee10PassTSR          - PME operates as 10PassTS-R

          The desired operational subtype of the PME can be configured
          via the efmCuPmeAdminSubType variable.

          If a Clause 45 MDIO Interface to the PMA/PMD is present, then
          this object combines values of the Port subtype select
          bits, the PMA/PMD type selection bits in the 10P/2B
          PMA/PMD control register, and the PMA/PMD link status bits in
          the 10P/2B PMA/PMD status register."
        REFERENCE
          "[802.3ah] 61.1, 45.2.1.11.4, 45.2.1.11.7, 45.2.1.12.4"
        ::= { efmCuPmeStatusEntry 3 }

      efmCuPmeOperProfile  OBJECT-TYPE
        SYNTAX      EfmProfileIndexOrZero
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "PME current operating profile.  This object is a pointer to
          an entry in either the efmCuPme2BProfileTable or the
          efmCuPme10PProfileTable, depending on the current operating
          SubType of the PME as indicated by efmCuPmeOperSubType.
          Note that a profile entry to which efmCuPmeOperProfile is
          pointing can be created automatically to reflect achieved
          parameters in adaptive (not fixed) initialization,
          i.e., values of efmCuPmeOperProfile and efmCuAdminProfile or
          efmCuPmeAdminProfile may differ.
          The value of zero indicates that the PME is Down or
          Initializing.

          This object partially maps to the aOperatingProfile attribute
          in Clause 30."
        REFERENCE
          "[802.3ah] 30.11.2.1.7"
        ::= { efmCuPmeStatusEntry 4 }
      efmCuPmeSnrMgn OBJECT-TYPE
        SYNTAX      Integer32(-127..128|65535)
        UNITS       "dB"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The current Signal to Noise Ratio (SNR) margin with respect
          to the received signal as perceived by the local PME.
          The value of 65535 is returned when the PME is Down or
          Initializing.

          This object maps to the aPMESNRMgn attribute in Clause 30.

          If a Clause 45 MDIO Interface is present, then this
          object maps to the 10P/2B RX SNR margin register."
        REFERENCE
          "[802.3ah] 30.11.2.1.4, 45.2.1.16"
        ::= { efmCuPmeStatusEntry 5 }

      efmCuPmePeerSnrMgn OBJECT-TYPE
        SYNTAX      Integer32(-127..128|65535)
        UNITS       "dB"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The current SNR margin in dB with respect to the received
          signal, as perceived by the remote (link partner) PME.
          The value of 65535 is returned when the PME is Down or
          Initializing.

          This object is irrelevant for the -R PME subtypes.  The value
          of 65535 SHALL be returned in this case.

          If a Clause 45 MDIO Interface is present, then this
          object maps to the 10P/2B link partner RX SNR margin
          register."
        REFERENCE
          "[802.3ah] 45.2.1.17"
        ::= { efmCuPmeStatusEntry 6}

      efmCuPmeLineAtn OBJECT-TYPE
        SYNTAX      Integer32(-127..128|65535)
        UNITS       "dB"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The current Line Attenuation in dB as perceived by the local
          PME.
          The value of 65535 is returned when the PME is Down or
          Initializing.

          If a Clause 45 MDIO Interface is present, then this
          object maps to the Line Attenuation register."
        REFERENCE
          "[802.3ah] 45.2.1.18"
        ::= { efmCuPmeStatusEntry 7 }

      efmCuPmePeerLineAtn OBJECT-TYPE
        SYNTAX      Integer32(-127..128|65535)
        UNITS       "dB"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The current Line Attenuation in dB as perceived by the remote
          (link partner) PME.
          The value of 65535 is returned when the PME is Down or
          Initializing.

          This object is irrelevant for the -R PME subtypes.  The value
          of 65535 SHALL be returned in this case.

          If a Clause 45 MDIO Interface is present, then this
          object maps to the 20P/2B link partner Line Attenuation
          register."
        REFERENCE
          "[802.3ah] 45.2.1.19"
        ::= { efmCuPmeStatusEntry 8 }

      efmCuPmeEquivalentLength  OBJECT-TYPE
        SYNTAX      Unsigned32(0..8192|65535)
        UNITS       "m"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "An estimate of the equivalent loop’s physical length in
          meters, as perceived by the PME after the link is established.
          An equivalent loop is a hypothetical 26AWG (0.4mm) loop with a
          perfect square root attenuation characteristic, without any
          bridged taps.
          The value of 65535 is returned if the link is Down or
          Initializing or the PME is unable to estimate the equivalent
          length.

          For a 10BASE-TL PME, if a Clause 45 MDIO Interface to the PME
          is present, then this object maps to the 10P Electrical Length
          register."
```

```
         REFERENCE
           "[802.3ah] 45.2.1.21"
         ::= { efmCuPmeStatusEntry 9 }

       efmCuPmeTCCodingErrors OBJECT-TYPE
         SYNTAX      Counter32
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
           "The number of 64/65-octet encapsulation errors.  This counter
           is incremented for each 64/65-octet encapsulation error
           detected by the 64/65-octet receive function.

           This object maps to aTCCodingViolations attribute in
           Clause 30.

           If a Clause 45 MDIO Interface to the PME TC is present, then
           this object maps to the TC coding violations register
           (see 45.2.6.12).

           Discontinuities in the value of this counter can occur at
           re-initialization of the management system, and at other times
           as indicated by the value of ifCounterDiscontinuityTime,
           defined in IF-MIB."
         REFERENCE
           "[802.3ah] 61.3.3.1, 30.11.2.1.5, 45.2.6.12"
         ::= { efmCuPmeStatusEntry 10 }

       efmCuPmeTCCrcErrors OBJECT-TYPE
         SYNTAX      Counter32
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
           "The number of TC-CRC errors.  This counter is incremented for
           each TC-CRC error detected by the 64/65-octet receive function
           (see 61.3.3.3 and Figure 61-19).

           This object maps to aTCCRCErrors attribute in
           Clause 30.

           If a Clause 45 MDIO Interface to the PME TC is present, then
           this object maps to the TC CRC error register
           (see 45.2.6.11).

           Discontinuities in the value of this counter can occur at
           re-initialization of the management system, and at other times
           as indicated by the value of ifCounterDiscontinuityTime,
           defined in IF-MIB."
```

```
        REFERENCE
          "[802.3ah] 61.3.3.3, 30.11.2.1.10, 45.2.6.11"
        ::= { efmCuPmeStatusEntry 11 }

     -- 2BASE-TL specific PME group

      efmCuPme2B      OBJECT IDENTIFIER ::= { efmCuPme 5 }

      efmCuPme2BProfileTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPme2BProfileEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "This table supports definitions of administrative and
          operating profiles for 2BASE-TL PMEs.
          The first 14 entries in this table SHALL always be defined as
          follows (see 802.3ah Annex 63A):
          -------+-------+-------+-----+------+------------+-----------
          Profile MinRate MaxRate Power Region Constellation Comment
           index  (Kbps)  (Kbps)  (dBm)
          -------+-------+-------+-----+------+------------+-----------
             1     5696    5696    13.5    1   32-TCPAM      default
             2     3072    3072    13.5    1   32-TCPAM
             3     2048    2048    13.5    1   16-TCPAM
             4     1024    1024    13.5    1   16-TCPAM
             5      704     704    13.5    1   16-TCPAM
             6      512     512    13.5    1   16-TCPAM
             7     5696    5696    14.5    2   32-TCPAM
             8     3072    3072    14.5    2   32-TCPAM
             9     2048    2048    14.5    2   16-TCPAM
            10     1024    1024    13.5    2   16-TCPAM
            11      704     704    13.5    2   16-TCPAM
            12      512     512    13.5    2   16-TCPAM
            13      192    5696       0    1   0             best effort
            14      192    5696       0    2   0             best effort
          -------+-------+-------+-----+------+------------+-----------

          These default entries SHALL be created during agent
          initialization and MUST NOT be deleted.

          Entries following the first 14 can be dynamically created and
          deleted to provide custom administrative (configuration)
          profiles and automatic operating profiles.

          This table MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] Annex 63A, 30.11.2.1.6"
        ::= { efmCuPme2B  2 }
      efmCuPme2BProfileEntry OBJECT-TYPE
        SYNTAX      EfmCuPme2BProfileEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Each entry corresponds to a single 2BASE-TL PME profile.
          Each profile contains a set of parameters, used either for
          configuration or representation of a 2BASE-TL PME.
          In case a particular profile is referenced via the
          efmCuPmeAdminProfile object (or efmCuAdminProfile if
          efmCuPmeAdminProfile is zero), it represents the desired
          parameters for the 2BaseTL-O PME initialization.
          If a profile is referenced via an efmCuPmeOperProfile object,
          it represents the current operating parameters of an
          operational PME.

          Profiles may be created/deleted using the row creation/
          deletion mechanism via efmCuPme2BProfileRowStatus.  If an
          active entry is referenced, the entry MUST remain 'active'
          until all references are removed.
          Default entries MUST NOT be removed."
        INDEX { efmCuPme2BProfileIndex }
        ::= { efmCuPme2BProfileTable 1 }

      EfmCuPme2BProfileEntry ::=
        SEQUENCE {
          efmCuPme2BProfileIndex           EfmProfileIndex,
          efmCuPme2BProfileDescr           SnmpAdminString,
          efmCuPme2BRegion                 INTEGER,
          efmCuPme2BsMode                  EfmProfileIndexOrZero,
          efmCuPme2BMinDataRate            Unsigned32,
          efmCuPme2BMaxDataRate            Unsigned32,
          efmCuPme2BPower                  Unsigned32,
          efmCuPme2BConstellation          INTEGER,
          efmCuPme2BProfileRowStatus       RowStatus
        }

      efmCuPme2BProfileIndex OBJECT-TYPE
        SYNTAX      EfmProfileIndex
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "2BASE-TL PME profile index.
          This object is the unique index associated with this profile.
          Entries in this table are referenced via efmCuAdminProfile or
          efmCuPmeAdminProfile objects."
        ::= { efmCuPme2BProfileEntry 1 }
      efmCuPme2BProfileDescr OBJECT-TYPE
        SYNTAX      SnmpAdminString
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "A textual string containing information about a 2BASE-TL PME
          profile.  The string may include information about the data
          rate and spectral limitations of this particular profile."
        ::= { efmCuPme2BProfileEntry 2 }

      efmCuPme2BRegion  OBJECT-TYPE
        SYNTAX      INTEGER {
          region1(1),
          region2(2)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Regional settings for a 2BASE-TL PME, as specified in the
          relevant Regional Annex of [G.991.2].
          Regional settings specify the Power Spectral Density (PSD)
          mask and the Power Back-Off (PBO) values, and place
          limitations on the max allowed data rate, power, and
          constellation.

          Possible values for this object are:
            region1      - Annexes A and F (e.g., North America)
            region2      - Annexes B and G (e.g., Europe)

          Annex A/B specify regional settings for data rates 192-2304
          Kbps using 16-TCPAM encoding.
          Annex F/G specify regional settings for rates 2320-3840 Kbps
          using 16-TCPAM encoding and 768-5696 Kbps using 32-TCPAM
          encoding.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object partially maps to the Region bits in the 2B general
          parameter register."
        REFERENCE
          "[802.3ah] 45.2.1.42; [G.991.2] Annexes A, B, F and G"
        ::= { efmCuPme2BProfileEntry 3 }

      efmCuPme2BsMode  OBJECT-TYPE
        SYNTAX      EfmProfileIndexOrZero
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Desired custom Spectral Mode for a 2BASE-TL PME.  This object
          is a pointer to an entry in efmCuPme2BsModeTable and a block
          of entries in efmCuPme2BRateReachTable, which together define
          (country-specific) reach-dependent rate limitations in
          addition to those defined by efmCuPme2BRegion.

          The value of this object is the index of the referenced
          spectral mode.
          The value of zero (default) indicates that no specific
          spectral mode is applicable.

          Attempts to set this object to a value that is not the value
          of the index for an active entry in the corresponding spectral
          mode table MUST be rejected."
        REFERENCE
          "efmCuPme2BsModeTable, efmCuPme2BRateReachTable"
        DEFVAL { 0 }
        ::= { efmCuPme2BProfileEntry 4 }

      efmCuPme2BMinDataRate  OBJECT-TYPE
        SYNTAX  Unsigned32(192..5696)
        UNITS       "Kbps"
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Minimum Data Rate for the 2BASE-TL PME.
          This object can take values of (n x 64)Kbps,
          where n=3..60 for 16-TCPAM and n=12..89 for 32-TCPAM encoding.

          The data rate of the 2BASE-TL PME is considered 'fixed' when
          the value of this object equals that of efmCuPme2BMaxDataRate.
          If efmCuPme2BMinDataRate is less than efmCuPme2BMaxDataRate in
          the administrative profile, the data rate is considered
          'adaptive', and SHALL be set to the maximum attainable rate
          not exceeding efmCuPme2BMaxDataRate, under the spectral
          limitations placed by the efmCuPme2BRegion and
          efmCuPme2BsMode.

          Note that the current operational data rate of the PME is
          represented by the ifSpeed object of IF-MIB.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object maps to the Min Data Rate1 bits in the 2B PMD
          parameters register.

          This object MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] 45.2.1.43"
        ::= { efmCuPme2BProfileEntry 5 }
      efmCuPme2BMaxDataRate  OBJECT-TYPE
        SYNTAX  Unsigned32(192..5696)
        UNITS       "Kbps"
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Maximum Data Rate for the 2BASE-TL PME.
          This object can take values of (n x 64)Kbps,
          where n=3..60 for 16-TCPAM and n=12..89 for 32-TCPAM encoding.

          The data rate of the 2BASE-TL PME is considered 'fixed' when
          the value of this object equals that of efmCuPme2BMinDataRate.
          If efmCuPme2BMinDataRate is less than efmCuPme2BMaxDataRate in
          the administrative profile, the data rate is considered
          'adaptive', and SHALL be set to the maximum attainable rate
          not exceeding efmCuPme2BMaxDataRate, under the spectral
          limitations placed by the efmCuPme2BRegion and
          efmCuPme2BsMode.

          Note that the current operational data rate of the PME is
          represented by the ifSpeed object of IF-MIB.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object maps to the Max Data Rate1 bits in the 2B PMD
          parameters register.

          This object MUST be maintained in a persistent manner."
        REFERENCE
          "[802.3ah] 45.2.1.43"
        ::= { efmCuPme2BProfileEntry 6 }

      efmCuPme2BPower  OBJECT-TYPE
        SYNTAX      Unsigned32(0|10..42)
        UNITS       "0.5 dBm"
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Signal Transmit Power.  Multiple of 0.5 dBm.
          The value of 0 in the administrative profile means that the
          signal transmit power is not fixed and SHALL be set to
          maximize the attainable rate, under the spectral limitations
          placed by the efmCuPme2BRegion and efmCuPme2BsMode.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object maps to the Power1 bits in the 2B PMD parameters
          register."
        REFERENCE
          "[802.3ah] 45.2.1.43"
        ::= { efmCuPme2BProfileEntry 7 }

      efmCuPme2BConstellation  OBJECT-TYPE
        SYNTAX      INTEGER {
          adaptive(0),
          tcpam16(1),
          tcpam32(2)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "TCPAM Constellation of the 2BASE-TL PME.
          The possible values are:
            adaptive(0)    - either 16- or 32-TCPAM
            tcpam16(1)     - 16-TCPAM
            tcpam32(2)     - 32-TCPAM

          The value of adaptive(0) in the administrative profile means
          that the constellation is not fixed and SHALL be set to
          maximize the attainable rate, under the spectral limitations
          placed by the efmCuPme2BRegion and efmCuPme2BsMode.

          If a Clause 45 MDIO Interface to the PME is present, then this
          object maps to the Constellation1 bits in the 2B general
          parameter register."
        REFERENCE
           "[802.3ah] 45.2.1.43"
        ::= { efmCuPme2BProfileEntry 8 }

      efmCuPme2BProfileRowStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "This object controls the creation, modification, or deletion
          of the associated entry in the efmCuPme2BProfileTable per the
          semantics of RowStatus.

          If an 'active' entry is referenced via efmCuAdminProfile or
          efmCuPmeAdminProfile instance(s), the entry MUST remain
          'active'.

          An 'active' entry SHALL NOT be modified.  In order to modify
          an existing entry, it MUST be taken out of service (by setting
          this object to 'notInService'), modified, and set 'active'
          again."
        ::= { efmCuPme2BProfileEntry 9 }
      efmCuPme2BsModeTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPme2BsModeEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "This table, together with efmCu2BReachRateTable, supports
          definition of administrative custom spectral modes for
          2BASE-TL PMEs, describing spectral limitations in addition to
          those specified by efmCuPme2BRegion.

          In some countries, spectral regulations (e.g., UK ANFP) limit
          the length of the loops for certain data rates.  This table
          allows these country-specific limitations to be specified.

          Entries in this table referenced by the efmCuPme2BsMode
          MUST NOT be deleted until all the active references are
          removed.

          This table MUST be maintained in a persistent manner."
        REFERENCE
          "efmCu2BReachRateTable"
        ::= { efmCuPme2B  3 }

      efmCuPme2BsModeEntry OBJECT-TYPE
        SYNTAX      EfmCuPme2BsModeEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Each entry specifies a spectral mode description and its
          index, which is used to reference corresponding entries in the
          efmCu2BReachRateTable.

          Entries may be created/deleted using the row creation/
          deletion mechanism via efmCuPme2BsModeRowStatus."
        INDEX { efmCuPme2BsModeIndex }
        ::= { efmCuPme2BsModeTable 1 }

      EfmCuPme2BsModeEntry ::=
        SEQUENCE {
          efmCuPme2BsModeIndex             EfmProfileIndex,
          efmCuPme2BsModeDescr             SnmpAdminString,
          efmCuPme2BsModeRowStatus         RowStatus
        }

      efmCuPme2BsModeIndex OBJECT-TYPE
        SYNTAX      EfmProfileIndex
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "2BASE-TL PME Spectral Mode index.
          This object is the unique index associated with this spectral
          mode.
          Entries in this table are referenced via the efmCuPme2BsMode
          object."
        ::= { efmCuPme2BsModeEntry 1 }

      efmCuPme2BsModeDescr OBJECT-TYPE
        SYNTAX      SnmpAdminString
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "A textual string containing information about a 2BASE-TL PME
          spectral mode.  The string may include information about
          corresponding (country-specific) spectral regulations
          and rate/reach limitations of this particular spectral mode."
        ::= { efmCuPme2BsModeEntry 2 }

      efmCuPme2BsModeRowStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "This object controls creation, modification, or deletion of
          the associated entry in efmCuPme2BsModeTable per the semantics
          of RowStatus.

          If an 'active' entry is referenced via efmCuPme2BsMode
          instance(s), the entry MUST remain 'active'.

          An 'active' entry SHALL NOT be modified.  In order to modify
          an existing entry, it MUST be taken out of service (by setting
          this object to 'notInService'), modified, and set 'active'
          again."
        ::= { efmCuPme2BsModeEntry 3 }


      efmCuPme2BReachRateTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPme2BReachRateEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "This table supports the definition of administrative custom
          spectral modes for 2BASE-TL PMEs, providing spectral
          limitations in addition to those specified by
          efmCuPme2BRegion.
          The spectral regulations in some countries (e.g., UK ANFP)
          limit the length of the loops for certain data rates.
          This table allows these country-specific limitations to be
          specified.

          Below is an example of this table for [ANFP]:
          ----------+-------+-------
          Equivalent MaxRate MaxRate
            Length    PAM16   PAM32
              (m)     (Kbps)  (Kbps)
          ----------+-------+-------
              975      2304    5696
             1125      2304    5504
             1275      2304    5120
             1350      2304    4864
             1425      2304    4544
             1500      2304    4288
             1575      2304    3968
             1650      2304    3776
             1725      2304    3520
             1800      2304    3264
             1875      2304    3072
             1950      2048    2688
             2100      1792    2368
             2250      1536       0
             2400      1408       0
             2550      1280       0
             2775      1152       0
             2925      1152       0
             3150      1088       0
             3375      1024       0
          ----------+-------+-------

          Entries in this table referenced by an efmCuPme2BsMode
          instance MUST NOT be deleted.

          This table MUST be maintained in a persistent manner."
        REFERENCE
          "[ANFP]"
        ::= { efmCuPme2B  4 }

      efmCuPme2BReachRateEntry OBJECT-TYPE
        SYNTAX      EfmCuPme2BReachRateEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Each entry specifies maximum 2BASE-TL PME data rates
          allowed for a certain equivalent loop length, when using
          16-TCPAM or 32-TCPAM encoding.

          When a 2BASE-TL PME is initialized, its data rate MUST NOT
          exceed one of the following limitations:
          - the value of efmCuPme2BMaxDataRate
          - maximum data rate allowed by efmCuPme2BRegion and
            efmCuPme2BPower
          - maximum data rate for a given encoding specified in the
            efmCuPme2BsModeEntry, corresponding to the equivalent loop
            length, estimated by the PME

          It is RECOMMENDED that the efmCuPme2BEquivalentLength values
          are assigned in increasing order, starting from the minimum
          value.

          Entries may be created/deleted using the row creation/
          deletion mechanism via efmCuPme2ReachRateRowStatus."
        INDEX { efmCuPme2BsModeIndex, efmCuPme2BReachRateIndex }
        ::= { efmCuPme2BReachRateTable 1 }

      EfmCuPme2BReachRateEntry ::=
        SEQUENCE {
          efmCuPme2BReachRateIndex         EfmProfileIndex,
          efmCuPme2BEquivalentLength       Unsigned32,
          efmCuPme2BMaxDataRatePam16       Unsigned32,
          efmCuPme2BMaxDataRatePam32       Unsigned32,
          efmCuPme2BReachRateRowStatus     RowStatus
        }

      efmCuPme2BReachRateIndex OBJECT-TYPE
        SYNTAX      EfmProfileIndex
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "2BASE-TL custom spectral mode Reach-Rate table index.
          This object is the unique index associated with each entry."
        ::= { efmCuPme2BReachRateEntry 1 }

      efmCuPme2BEquivalentLength  OBJECT-TYPE
        SYNTAX      Unsigned32(0..8192)
        UNITS       "m"
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Maximum allowed equivalent loop’s physical length in meters
          for the specified data rates.
          An equivalent loop is a hypothetical 26AWG (0.4mm) loop with a
          perfect square root attenuation characteristic, without any
          bridged taps."
        ::= { efmCuPme2BReachRateEntry 2 }

      efmCuPme2BMaxDataRatePam16  OBJECT-TYPE
        SYNTAX      Unsigned32(0|192..5696)
        UNITS       "Kbps"
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Maximum data rate for a 2BASE-TL PME at the specified
          equivalent loop's length using TC-PAM16 encoding.
          The value of zero means that TC-PAM16 encoding should not be
          used at this distance."
        ::= { efmCuPme2BReachRateEntry 3 }

      efmCuPme2BMaxDataRatePam32  OBJECT-TYPE
        SYNTAX      Unsigned32(0|192..5696)
        UNITS       "Kbps"
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "Maximum data rate for a 2BASE-TL PME at the specified
          equivalent loop's length using TC-PAM32 encoding.
          The value of zero means that TC-PAM32 encoding should not be
          used at this distance."
        ::= { efmCuPme2BReachRateEntry 4 }

      efmCuPme2BReachRateRowStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "This object controls the creation, modification, or deletion
          of the associated entry in the efmCuPme2BReachRateTable per
          the semantics of RowStatus.

          If an 'active' entry is referenced via efmCuPme2BsMode
          instance(s), the entry MUST remain 'active'.

          An 'active' entry SHALL NOT be modified.  In order to modify
          an existing entry, it MUST be taken out of service (by setting
          this object to 'notInService'), modified, and set 'active'
          again."
        ::= { efmCuPme2BReachRateEntry 5 }


     -- 10PASS-TS specific PME group
```

```
      efmCuPme10P      OBJECT IDENTIFIER ::= { efmCuPme 6 }

      efmCuPme10PProfileTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPme10PProfileEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "This table supports definitions of configuration profiles for
          10PASS-TS PMEs.
          The first 22 entries in this table SHALL always be defined as
          follows (see 802.3ah Annex 62B.3, table 62B-1):
          -------+--------+----+---------+-----+-----+---------------
          Profile Bandplan UPBO BandNotch DRate URate Comment
           Index  PSDMask#  p#    p#        p#    p#
          -------+--------+----+---------+-----+-----+---------------
             1      1      3    2,6,10,11    20    20 default profile
             2     13      5    0            20    20
             3      1      1    0            20    20
             4     16      0    0           100   100
             5     16      0    0            70    50
             6      6      0    0            50    10
             7     17      0    0            30    30
             8      8      0    0            30     5
             9      4      0    0            25    25
            10      4      0    0            15    15
            11     23      0    0            10    10
            12     23      0    0             5     5
            13     16      0    2,5,9,11    100   100
            14     16      0    2,5,9,11     70    50
            15      6      0    2,6,10,11    50    10
            16     17      0    2,5,9,11     30    30
            17      8      0    2,6,10,11    30     5
            18      4      0    2,6,10,11    25    25
            19      4      0    2,6,10,11    15    15
            20     23      0    2,5,9,11     10    10
            21     23      0    2,5,9,11      5     5
            22     30      0    0           200    50
          -------+--------+----+---------+-----+-----+---------------

          These default entries SHALL be created during agent
          initialization and MUST NOT be deleted.

          Entries following the first 22 can be dynamically created and
          deleted to provide custom administrative (configuration)
          profiles and automatic operating profiles.

          This table MUST be maintained in a persistent manner."
        REFERENCE
```

```
          "[802.3ah] Annex 62B.3, 30.11.2.1.6"
        ::= { efmCuPme10P  1 }

      efmCuPme10PProfileEntry OBJECT-TYPE
        SYNTAX      EfmCuPme10PProfileEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "Each entry corresponds to a single 10PASS-TS PME profile.

          Each profile contains a set of parameters, used either for
          configuration or representation of a 10PASS-TS PME.
          In case a particular profile is referenced via the
          efmCuPmeAdminProfile object (or efmCuAdminProfile if
          efmCuPmeAdminProfile is zero), it represents the desired
          parameters for the 10PassTS-O PME initialization.
          If a profile is referenced via an efmCuPmeOperProfile object,
          it represents the current operating parameters of the PME.

          Profiles may be created/deleted using the row creation/
          deletion mechanism via efmCuPme10PProfileRowStatus.  If an
          'active' entry is referenced, the entry MUST remain 'active'
          until all references are removed.
          Default entries MUST NOT be removed."
        INDEX { efmCuPme10PProfileIndex }
        ::= { efmCuPme10PProfileTable 1 }

      EfmCuPme10PProfileEntry ::=
        SEQUENCE {
          efmCuPme10PProfileIndex           EfmProfileIndex,
          efmCuPme10PProfileDescr           SnmpAdminString,
          efmCuPme10PBandplanPSDMskProfile  INTEGER,
          efmCuPme10PUPBOReferenceProfile   INTEGER,
          efmCuPme10PBandNotchProfiles      BITS,
          efmCuPme10PPayloadDRateProfile    INTEGER,
          efmCuPme10PPayloadURateProfile    INTEGER,
          efmCuPme10PProfileRowStatus       RowStatus
        }

      efmCuPme10PProfileIndex OBJECT-TYPE
        SYNTAX      EfmProfileIndex
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "10PASS-TS PME profile index.
          This object is the unique index associated with this profile.
          Entries in this table are referenced via efmCuAdminProfile or
          efmCuPmeAdminProfile."
```

```
         ::= { efmCuPme10PProfileEntry 1 }

       efmCuPme10PProfileDescr OBJECT-TYPE
         SYNTAX      SnmpAdminString
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
           "A textual string containing information about a 10PASS-TS PME
           profile.  The string may include information about data rate
           and spectral limitations of this particular profile."
         ::= { efmCuPme10PProfileEntry 2 }

       efmCuPme10PBandplanPSDMskProfile  OBJECT-TYPE
         SYNTAX  INTEGER {
           profile1(1),
           profile2(2),
           profile3(3),
           profile4(4),
           profile5(5),
           profile6(6),
           profile7(7),
           profile8(8),
           profile9(9),
           profile10(10),
           profile11(11),
           profile12(12),
           profile13(13),
           profile14(14),
           profile15(15),
           profile16(16),
           profile17(17),
           profile18(18),
           profile19(19),
           profile20(20),
           profile21(21),
           profile22(22),
           profile23(23),
           profile24(24),
           profile25(25),
           profile26(26),
           profile27(27),
           profile28(28),
           profile29(29),
           profile30(30)
         }
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
```

```
          "The 10PASS-TS PME Bandplan and PSD Mask Profile, as specified
          in 802.3ah Annex 62A, table 62A-1.  Possible values are:
          --------------+------------------------+-----------+--------
          Profile Name    PSD Mask                  Bands      G.993.1
                                                  0/1/2/3/4/5  Bandplan
          --------------+------------------------+-----------+--------
          profile1(1)    T1.424 FTTCab.M1         x/D/U/D/U    A
          profile2(2)    T1.424 FTTEx.M1          x/D/U/D/U    A
          profile3(3)    T1.424 FTTCab.M2         x/D/U/D/U    A
          profile4(4)    T1.424 FTTEx.M2          x/D/U/D/U    A
          profile5(5)    T1.424 FTTCab.M1         D/D/U/D/U    A
          profile6(6)    T1.424 FTTEx.M1          D/D/U/D/U    A
          profile7(7)    T1.424 FTTCab.M2         D/D/U/D/U    A
          profile8(8)    T1.424 FTTEx.M2          D/D/U/D/U    A
          profile9(9)    T1.424 FTTCab.M1         U/D/U/D/x    A
          profile10(10)  T1.424 FTTEx.M1          U/D/U/D/x    A
          profile11(11)  T1.424 FTTCab.M2         U/D/U/D/x    A
          profile12(12)  T1.424 FTTEx.M2          U/D/U/D/x    A
          profile13(13)  TS 101 270-1 Pcab.M1.A   x/D/U/D/U    B
          profile14(14)  TS 101 270-1 Pcab.M1.B   x/D/U/D/U    B
          profile15(15)  TS 101 270-1 Pex.P1.M1   x/D/U/D/U    B
          profile16(16)  TS 101 270-1 Pex.P2.M1   x/D/U/D/U    B
          profile17(17)  TS 101 270-1 Pcab.M2     x/D/U/D/U    B
          profile18(18)  TS 101 270-1 Pex.P1.M2   x/D/U/D/U    B
          profile19(19)  TS 101 270-1 Pex.P2.M2   x/D/U/D/U    B
          profile20(20)  TS 101 270-1 Pcab.M1.A   U/D/U/D/x    B
          profile21(21)  TS 101 270-1 Pcab.M1.B   U/D/U/D/x    B
          profile22(22)  TS 101 270-1 Pex.P1.M1   U/D/U/D/x    B
          profile23(23)  TS 101 270-1 Pex.P2.M1   U/D/U/D/x    B
          profile24(24)  TS 101 270-1 Pcab.M2     U/D/U/D/x    B
          profile25(25)  TS 101 270-1 Pex.P1.M2   U/D/U/D/x    B
          profile26(26)  TS 101 270-1 Pex.P2.M2   U/D/U/D/x    B
          profile27(27)  G.993.1 F.1.2.1          x/D/U/D/U    Annex F
          profile28(28)  G.993.1 F.1.2.2          x/D/U/D/U    Annex F
          profile29(29)  G.993.1 F.1.2.3          x/D/U/D/U    Annex F
          profile30(30)  T1.424 FTTCab.M1 (ext.)  x/D/U/D/U/D  Annex A
          --------------+------------------------+-----------+--------
          "
        REFERENCE
          "[802.3ah] Annex 62A"
        ::= { efmCuPme10PProfileEntry 3 }

      efmCuPme10PUPBOReferenceProfile  OBJECT-TYPE
        SYNTAX  INTEGER {
          profile0(0),
          profile1(1),
          profile2(2),
          profile3(3),
          profile4(4),
          profile5(5),
          profile6(6),
          profile7(7),
          profile8(8),
          profile9(9)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "The 10PASS-TS PME Upstream Power Back-Off (UPBO) Reference
          PSD Profile, as specified in 802.3 Annex 62A, table 62A-3.
          Possible values are:
          ------------+-----------------------------
          Profile Name   Reference PSD
          ------------+-----------------------------
          profile0(0)  no profile
          profile1(1)  T1.424        Noise A    M1
          profile2(2)  T1.424        Noise A    M2
          profile3(3)  T1.424        Noise F    M1
          profile4(4)  T1.424        Noise F    M2
          profile5(5)  TS 101 270-1  Noise A&B
          profile6(6)  TS 101 270-1  Noise C
          profile7(7)  TS 101 270-1  Noise D
          profile8(8)  TS 101 270-1  Noise E
          profile9(9)  TS 101 270-1  Noise F
          ------------+-----------------------------
          "
        REFERENCE
          "[802.3ah] Annex 62A.3.5"
        ::= { efmCuPme10PProfileEntry 4 }

      efmCuPme10PBandNotchProfiles  OBJECT-TYPE
        SYNTAX  BITS {
          profile0(0),
          profile1(1),
          profile2(2),
          profile3(3),
          profile4(4),
          profile5(5),
          profile6(6),
          profile7(7),
          profile8(8),
          profile9(9),
          profile10(10),
          profile11(11)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "The 10PASS-TS PME Egress Control Band Notch Profile bitmap,
          as specified in 802.3 Annex 62A, table 62A-4.  Possible values
          are:
          --------------+--------+------+------------+------+------
          Profile Name   G.991.3  T1.424 TS 101 270-1 StartF EndF
                         table    table  table        (MHz)  (MHz)
          --------------+--------+------+------------+------+------
          profile0(0)    no profile
          profile1(1)    F-5 #01  -      -            1.810  1.825
          profile2(2)    6-2      15-1   17           1.810  2.000
          profile3(3)    F-5 #02  -      -            1.907  1.912
          profile4(4)    F-5 #03  -      -            3.500  3.575
          profile5(5)    6-2      -      17           3.500  3.800
          profile6(6)    -        15-1   -            3.500  4.000
          profile7(7)    F-5 #04  -      -            3.747  3.754
          profile8(8)    F-5 #05  -      -            3.791  3.805
          profile9(9)    6-2      -      17           7.000  7.100
          profile10(10)  F-5 #06  15-1   -            7.000  7.300
          profile11(11)  6-2      15-1   1            10.100 10.150
          --------------+--------+------+------------+------+------

          Any combination of profiles can be specified by ORing
          individual profiles, for example, a value of 0x2230 selects
          profiles 2, 6, 10, and 11."
        REFERENCE
          "[802.3ah] Annex 62A.3.5"
        ::= { efmCuPme10PProfileEntry 5 }

      efmCuPme10PPayloadDRateProfile  OBJECT-TYPE
        SYNTAX      INTEGER {
          profile5(5),
          profile10(10),
          profile15(15),
          profile20(20),
          profile25(25),
          profile30(30),
          profile50(50),
          profile70(70),
          profile100(100),
          profile140(140),
          profile200(200)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "The 10PASS-TS PME Downstream Payload Rate Profile, as
          specified in 802.3 Annex 62A.  Possible values are:
            profile5(5)      - 2.5 Mbps
            profile10(10)    - 5 Mbps
            profile15(15)    - 7.5 Mbps
            profile20(20)    - 10 Mbps
            profile25(25)    - 12.5 Mbps
            profile30(30)    - 15 Mbps
            profile50(50)    - 25 Mbps
            profile70(70)    - 35 Mbps
            profile100(100)  - 50 Mbps
            profile140(140)  - 70 Mbps
            profile200(200)  - 100 Mbps

          Each value represents a target for the PME’s Downstream
          Payload Bitrate as seen at the MII.  If the payload rate of
          the selected profile cannot be achieved based on the loop
          environment, bandplan, and PSD mask, the PME initialization
          SHALL fail."
        REFERENCE
          "[802.3ah] Annex 62A.3.6"
        ::= { efmCuPme10PProfileEntry 6 }

      efmCuPme10PPayloadURateProfile  OBJECT-TYPE
        SYNTAX      INTEGER {
          profile5(5),
          profile10(10),
          profile15(15),
          profile20(20),
          profile25(25),
          profile30(30),
          profile50(50),
          profile70(70),
          profile100(100)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "The 10PASS-TS PME Upstream Payload Rate Profile, as specified
           in 802.3 Annex 62A.  Possible values are:
            profile5(5)       - 2.5 Mbps
            profile10(10)     - 5 Mbps
            profile15(15)     - 7.5 Mbps
            profile20(20)     - 10 Mbps
            profile25(25)     - 12.5 Mbps
            profile30(30)     - 15 Mbps
            profile50(50)     - 25 Mbps
            profile70(70)     - 35 Mbps
            profile100(100)   - 50 Mbps
          Each value represents a target for the PME’s Upstream Payload
          Bitrate as seen at the MII.  If the payload rate of the
          selected profile cannot be achieved based on the loop
          environment, bandplan, and PSD mask, the PME initialization
          SHALL fail."
        REFERENCE
          "[802.3ah] Annex 62A.3.6"
        ::= { efmCuPme10PProfileEntry 7 }

      efmCuPme10PProfileRowStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
          "This object controls creation, modification, or deletion of
          the associated entry in efmCuPme10PProfileTable per the
          semantics of RowStatus.

          If an active entry is referenced via efmCuAdminProfile or
          efmCuPmeAdminProfile, the entry MUST remain ’active’ until
          all references are removed.

          An ’active’ entry SHALL NOT be modified.  In order to modify
          an existing entry, it MUST be taken out of service (by setting
          this object to ’notInService’), modified, and set ’active’
          again."
        ::= { efmCuPme10PProfileEntry 8 }


      efmCuPme10PStatusTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF EfmCuPme10PStatusEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "This table provides status information of EFMCu 10PASS-TS
          PMEs (modems).

          This table contains live data from the equipment.  As such,
          it is NOT persistent."
        ::= { efmCuPme10P 2 }

      efmCuPme10PStatusEntry OBJECT-TYPE
        SYNTAX      EfmCuPme10PStatusEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "An entry in the EFMCu 10PASS-TS PME Status table."
        INDEX  { ifIndex }
        ::= { efmCuPme10PStatusTable 1 }

      EfmCuPme10PStatusEntry ::=
        SEQUENCE {
          efmCuPme10PFECCorrectedBlocks     Counter32,
          efmCuPme10PFECUncorrectedBlocks   Counter32
        }

      efmCuPme10PFECCorrectedBlocks  OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of received and corrected Forward Error Correction
          (FEC) codewords in this 10PASS-TS PME.

          This object maps to the aPMEFECCorrectedBlocks attribute in
          Clause 30.

          If a Clause 45 MDIO Interface to the PMA/PMD is present,
          then this object maps to the 10P FEC correctable errors
          register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.1.22, 30.11.2.1.8"
        ::= { efmCuPme10PStatusEntry 1 }

      efmCuPme10PFECUncorrectedBlocks  OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of received uncorrectable FEC codewords in this
          10PASS-TS PME.

          This object maps to the aPMEFECUncorrectableBlocks attribute
          in Clause 30.

          If a Clause 45 MDIO Interface to the PMA/PMD is present,
          then this object maps to the 10P FEC uncorrectable errors
          register.

          Discontinuities in the value of this counter can occur at
          re-initialization of the management system, and at other times
          as indicated by the value of ifCounterDiscontinuityTime,
          defined in IF-MIB."
        REFERENCE
          "[802.3ah] 45.2.1.23, 30.11.2.1.9"
        ::= { efmCuPme10PStatusEntry 2 }

   --
   -- Conformance Statements
   --

      efmCuGroups      OBJECT IDENTIFIER ::= { efmCuConformance 1 }

      efmCuCompliances OBJECT IDENTIFIER ::= { efmCuConformance 2 }

      -- Object Groups

      efmCuBasicGroup OBJECT-GROUP
        OBJECTS {
          efmCuPAFSupported,
          efmCuAdminProfile,
          efmCuTargetDataRate,
          efmCuTargetSnrMgn,
          efmCuAdaptiveSpectra,
          efmCuPortSide,
          efmCuFltStatus
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects representing management information
          common for all types of EFMCu ports."
        ::= { efmCuGroups 1 }

      efmCuPAFGroup OBJECT-GROUP
        OBJECTS {
          efmCuPeerPAFSupported,
          efmCuPAFCapacity,
          efmCuPeerPAFCapacity,
          efmCuPAFAdminState,
          efmCuPAFDiscoveryCode,
          efmCuPAFRemoteDiscoveryCode,
          efmCuNumPMEs
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects supporting OPTIONAL PME
          Aggregation Function (PAF) and PAF discovery in EFMCu ports."
        ::= { efmCuGroups 2 }
      efmCuPAFErrorsGroup OBJECT-GROUP
        OBJECTS {
          efmCuPAFInErrors,
          efmCuPAFInSmallFragments,
          efmCuPAFInLargeFragments,
          efmCuPAFInBadFragments,
          efmCuPAFInLostFragments,
          efmCuPAFInLostStarts,
          efmCuPAFInLostEnds,
          efmCuPAFInOverflows
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects supporting OPTIONAL error counters
          of PAF on EFMCu ports."
        ::= { efmCuGroups 3 }

      efmCuPmeGroup OBJECT-GROUP
        OBJECTS {
          efmCuPmeAdminProfile,
          efmCuPmeOperStatus,
          efmCuPmeFltStatus,
          efmCuPmeSubTypesSupported,
          efmCuPmeAdminSubType,
          efmCuPmeOperSubType,
          efmCuPAFRemoteDiscoveryCode,
          efmCuPmeOperProfile,
          efmCuPmeSnrMgn,
          efmCuPmePeerSnrMgn,
          efmCuPmeLineAtn,
          efmCuPmePeerLineAtn,
          efmCuPmeEquivalentLength,
          efmCuPmeTCCodingErrors,
          efmCuPmeTCCrcErrors,
          efmCuPmeThreshLineAtn,
          efmCuPmeThreshSnrMgn
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects providing information about
          a 2BASE-TL/10PASS-TS PME."
        ::= { efmCuGroups 4 }

      efmCuAlarmConfGroup OBJECT-GROUP
        OBJECTS {
          efmCuThreshLowRate,
          efmCuLowRateCrossingEnable,
          efmCuPmeThreshLineAtn,
          efmCuPmeLineAtnCrossingEnable,
          efmCuPmeThreshSnrMgn,
          efmCuPmeSnrMgnCrossingEnable,
          efmCuPmeDeviceFaultEnable,
          efmCuPmeConfigInitFailEnable,
          efmCuPmeProtocolInitFailEnable
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects supporting configuration of alarm
          thresholds and notifications in EFMCu ports."
        ::= { efmCuGroups 5 }

      efmCuNotificationGroup NOTIFICATION-GROUP
        NOTIFICATIONS {
          efmCuLowRateCrossing,
          efmCuPmeLineAtnCrossing,
          efmCuPmeSnrMgnCrossing,
          efmCuPmeDeviceFault,
          efmCuPmeConfigInitFailure,
          efmCuPmeProtocolInitFailure
        }
        STATUS      current
        DESCRIPTION
          "This group supports notifications of significant conditions
          associated with EFMCu ports."
        ::= { efmCuGroups 6 }

      efmCuPme2BProfileGroup OBJECT-GROUP
        OBJECTS {
          efmCuPme2BProfileDescr,
          efmCuPme2BRegion,
          efmCuPme2BsMode,
          efmCuPme2BMinDataRate,
          efmCuPme2BMaxDataRate,
          efmCuPme2BPower,
          efmCuPme2BConstellation,
          efmCuPme2BProfileRowStatus,
          efmCuPme2BsModeDescr,
          efmCuPme2BsModeRowStatus,
          efmCuPme2BEquivalentLength,
          efmCuPme2BMaxDataRatePam16,
          efmCuPme2BMaxDataRatePam32,
          efmCuPme2BReachRateRowStatus
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects that constitute a configuration
          profile for configuration of 2BASE-TL ports."
        ::= { efmCuGroups 7}

      efmCuPme10PProfileGroup OBJECT-GROUP
        OBJECTS {
          efmCuPme10PProfileDescr,
          efmCuPme10PBandplanPSDMskProfile,
          efmCuPme10PUPBOReferenceProfile,
          efmCuPme10PBandNotchProfiles,
          efmCuPme10PPayloadDRateProfile,
          efmCuPme10PPayloadURateProfile,
          efmCuPme10PProfileRowStatus
        }
        STATUS  current
        DESCRIPTION
          "A collection of objects that constitute a configuration
          profile for configuration of 10PASS-TS ports."
        ::= { efmCuGroups 8 }

      efmCuPme10PStatusGroup OBJECT-GROUP
        OBJECTS {
          efmCuPme10PFECCorrectedBlocks,
          efmCuPme10PFECUncorrectedBlocks
        }
        STATUS  current
        DESCRIPTION
          "A collection of objects providing status information
          specific to 10PASS-TS PMEs."
        ::= { efmCuGroups 9 }

     -- Compliance Statements

      efmCuCompliance MODULE-COMPLIANCE
        STATUS      current
        DESCRIPTION
          "The compliance statement for 2BASE-TL/10PASS-TS interfaces.
          Compliance with the following external compliance statements
          is REQUIRED:

          MIB Module             Compliance Statement
          ----------             --------------------
          IF-MIB                 ifCompliance3
          EtherLike-MIB          dot3Compliance2
          MAU-MIB                mauModIfCompl3

          Compliance with the following external compliance statements
          is OPTIONAL for implementations supporting PME Aggregation
          Function (PAF) with flexible cross-connect between the PCS
          and PME ports:

          MIB Module              Compliance Statement
          ----------              --------------------
          IF-INVERTED-STACK-MIB  ifInvCompliance
          IF-CAP-STACK-MIB        ifCapStackCompliance"

        MODULE  -- this module
          MANDATORY-GROUPS {
            efmCuBasicGroup,
            efmCuPmeGroup,
            efmCuAlarmConfGroup,
            efmCuNotificationGroup
          }

          GROUP       efmCuPme2BProfileGroup
          DESCRIPTION
            "Support for this group is only required for implementations
            supporting 2BASE-TL PHY."

          GROUP       efmCuPme10PProfileGroup
          DESCRIPTION
            "Support for this group is only required for implementations
            supporting 10PASS-TS PHY."

          GROUP       efmCuPAFGroup
          DESCRIPTION
            "Support for this group is only required for
            implementations supporting PME Aggregation Function (PAF)."

          GROUP       efmCuPAFErrorsGroup
          DESCRIPTION
            "Support for this group is OPTIONAL for implementations
            supporting PME Aggregation Function (PAF)."

          GROUP       efmCuPme10PStatusGroup
          DESCRIPTION
            "Support for this group is OPTIONAL for implementations
            supporting 10PASS-TS PHY."

          OBJECT      efmCuPmeSubTypesSupported
          SYNTAX      BITS {
            ieee2BaseTLO(0),
            ieee2BaseTLR(1),
            ieee10PassTSO(2),
            ieee10PassTSR(3)
          }
          DESCRIPTION
```

```
            "Support for all subtypes is not required.  However, at
            least one value SHALL be supported."

          OBJECT      efmCuPmeAdminSubType
          MIN-ACCESS  read-only
          DESCRIPTION
            "Write access is not required (needed only for PMEs
            supporting more than a single subtype, e.g.,
            ieee2BaseTLO and ieee2BaseTLR or ieee10PassTSO and
            ieee10PassTSR)."

          OBJECT      efmCuTargetSnrMgn
          MIN-ACCESS  read-only
          DESCRIPTION
            "Write access is OPTIONAL.  For PHYs without write access,
            the target SNR margin SHALL be fixed at 5dB for 2BASE-TL
            and 6dB for 10PASS-TS."

          OBJECT      efmCuAdaptiveSpectra
          MIN-ACCESS  read-only
          DESCRIPTION
            "Write access is OPTIONAL.  For PHYs without write access,
            the default value SHOULD be false."

        ::= { efmCuCompliances 1 }
   END
```

7. Security Considerations

There is a number of managed objects defined in the EFM-CU-MIB module that have a MAX-ACCESS clause of read-write or read-create. Most objects are writeable only when the link is Down. Writing to these objects can have potentially disruptive effects on network operation, for example:

- Changing of efmCuPmeAdminSubType may lead to a potential locking of the link, as peer PMEs of the same subtype cannot exchange handshake messages.

- Changing of efmCuPAFAdminState to enabled may lead to a potential locking of the link, if the peer PHY does not support PAF.

- Changing of efmCuPAFDiscoveryCode, before the discovery operation, may lead to a wrongful discovery, for example, when two -O ports are connected to the same multi-PME -R port and both -O ports have the same Discovery register value.

- Changing PCS or PME configuration parameters (e.g., profile of a PCS or PME via efmCuAdminProfile or efmCuPmeAdminProfile) may lead to anything from link quality and rate degradation to a complete link initialization failure, as ability of an EFMCu port to support a particular configuration depends on the copper environment.

- Activation of a PME can cause a severe degradation of service for another EFMCu PHY, whose PME(s) may be affected by the cross-talk from the newly activated PME.

- Removal of a PME from an operationally ’up’ EFMCu port, aggregating several PMEs, may cause port’s rate degradation.

The user of the EFM-CU-MIB module must therefore be aware that support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations.

The readable objects in the EFM-CU-MIB module (i.e., those with MAX-ACCESS other than not-accessible) may be considered sensitive in some environments since, collectively, they provide information about the performance of network interfaces and can reveal some aspects of their configuration. In particular, since EFMCu can be carried over Unshielded Twisted Pair (UTP) voice-grade copper in a bundle with other pairs belonging to another operator/customer, it is theoretically possible to eavesdrop to an EFMCu transmission simply by "listening" to a cross-talk from the EFMCu pairs, especially if the parameters of the EFMCu link in question are known.

In such environments, it is important to control also GET and NOTIFY access to these objects and possibly even to encrypt their values when sending them over the network via SNMP.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in these MIB modules.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of these MIB modules is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

8. IANA Considerations

Object identifiers for the efmCuMIB MODULE-IDENTITY and ifCapStackMIB MODULE-IDENTITY have been allocated by IANA in the MIB-2 sub-tree.

9. Acknowledgments

This document was produced by the [HUBMIB] working group, whose efforts were greatly advanced by the contributions of the following people (in alphabetical order):

Udi Ashkenazi (Actelis)

Mike Heard

Alfred Hoenes (TR-Sys)

Marina Popilov (Actelis)

Mathias Riess (Infineon)

Dan Romascanu (Avaya)

Matt Squire (Hatteras)

Bert Wijnen (Alcatel)

10. References

10.1. Normative References

[802.3] IEEE, "IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications", IEEE Std 802.3-2005, December 2005.

[802.3ah]	IEEE, "IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications - Amendment: Media Access Control Parameters, Physical Layers and Management Parameters for Subscriber Access Networks", IEEE Std 802.3ah-2004, September 2004.
[G.991.2]	ITU-T, "Single-pair High-speed Digital Subscriber Line (SHDSL) transceivers", ITU-T Recommendation G.991.2, December 2003, <http://www.itu.int/rec/T-REC-G.991.2/en>.
[G.993.1]	ITU-T, "Very High speed Digital Subscriber Line transceivers", ITU-T Recommendation G.993.1, June 2004, <http://www.itu.int/rec/T-REC-G.993.1/en>.
[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
[RFC2578]	McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
[RFC2579]	McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
[RFC2580]	McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
[RFC2863]	McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.
[RFC2864]	McCloghrie, K. and G. Hanson, "The Inverted Stack Table Extension to the Interfaces Group MIB", RFC 2864, June 2000.
[RFC3411]	Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.

[RFC3635]	Flick, J., "Definitions of Managed Objects for the Ethernet-like Interface Types", RFC 3635, September 2003.
[RFC4836]	Beili, E., "Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)", RFC 4836, April 2007.
[T1.424]	ANSI, "Interface Between Networks and Customer Installation Very-high-bit-rate Digital Subscriber Lines (VDSL) Metallic Interface (DMT Based)", American National Standard T1.424-2004, June 2004.
[TS 101 270-1]	ETSI, "Transmission and Multiplexing (TM); Access transmission systems on metallic access cables; Very high speed Digital Subscriber Line (VDSL); Part 1: Functional requirements", Technical Specification TS 101 270-1, October 2005.

10.2. Informative References

[ANFP]	Network Interoperability Consultative Committee (NICC), "Specification of the Access Network Frequency Plan (ANFP) applicable to transmission systems used on the BT Access Network", NICC Document ND1602:2005/08, August 2005.
[HUBMIB]	IETF, "Ethernet Interfaces and Hub MIB (hubmib) Charter", <http://www.ietf.org/html.charters/OLD/hubmib-charter.html>.
[IANAifType-MIB]	Internet Assigned Numbers Authority (IANA), "IANAifType Textual Convention definition", <http://www.iana.org/assignments/ianaiftype-mib>.
[RFC3410]	Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.
[RFC4070]	Dodge, M. and B. Ray, "Definitions of Managed Object Extensions for Very High Speed Digital Subscriber Lines (VDSL) Using Multiple Carrier Modulation (MCM) Line Coding", RFC 4070, May 2005.
[RFC4181]	Heard, C., "Guidelines for Authors and Reviewers of MIB Documents", BCP 111, RFC 4181, September 2005.

[RFC4319]	Sikes, C., Ray, B., and R. Abbi, "Definitions of Managed Objects for High Bit-Rate DSL - 2nd generation (HDSL2) and Single-Pair High-Speed Digital Subscriber Line (SHDSL) Lines", RFC 4319, December 2005.
[RFC4837]	Khermosh, L., "Managed Objects of Ethernet Passive Optical Networks (EPON)", RFC 4837, July 2007.
[RFC4878]	Squire, M., "Definitions and Managed Objects for Operations, Administration, and Maintenance (OAM) Functions on Ethernet-Like Interfaces", RFC 4878, June 2007.

Author’s Address

Edward Beili
Actelis Networks
Bazel 25
Petach-Tikva
Israel

Phone: +972-3-924-3491
EMail: edward.beili@actelis.com

Full Copyright Statement

Intellectual Property