          Ethernet in the First Mile Copper (EFMCu) Interfaces MIB
                   draft-ietf-hubmib-efm-cu-mib-01.txt

Status of this Memo

Copyright Notice

Abstract

   This document defines a portion of the Management Information Base
   (MIB) for use with network management protocols in TCP/IP based
   nternets. This document proposes an extension to the Ethernet-like
   Interfaces MIB and MAU MIB with a set of objects for managing an
   Ethernet in the First Mile Copper (EFMCu) interfaces 10Pass-TS and
   2Base-TL, defined in IEEE standard 802.3ah.

Table of Contents

1. Introduction

   New Ethernet like interfaces have been defined in the Institute of
   Electrical and Electronics Engineers (IEEE) 802.3ah project a.k.a.
   Ethernet in the First Mile (EFM) [802.3ah]. 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 EFMCu, are based
   on ITU-T G.SHDSL and VDSL specifications and support variable rates
   and optional Physical Medium Entity (PME) aggregation (a.k.a.
   multi-pair bonding).

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

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

   This memo defines a portion of the Management Information Base (MIB)
   for use with network management protocols in the Internet community
   to manage EFMCu interfaces.

   Note that managed objects for Operation, Administration and
   Management (OAM) and Ethernet over Passive Optical Networks (EPON)
   clauses of IEEE 802.3ah are defined in EFM-COMMON-MIB
   [I-D.ietf-hubmib-efm-mib] and EFM-EPON-MIB
   [I-D.ietf-hubmib-efm-epon-mib] 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 a MIB
   module that is compliant to the SMIv2, which is described in STD 58,
   RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580
   [RFC2580]. A detailed introduction to the current SNMP Management
   Framework can be found in RFC 2570 [RFC2570].

   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 MIBs

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

3.1 Relation to Interfaces Group MIB

2Base-TL and 10Pass-TS PHY's specified in this MIB are stacked
Ethernet interfaces and as such are managed using generic interface
management objects defined in the IF-MIB [RFC2863]. The stack
management in done via the ifStackTable, as defined in the IF-MIB
[RFC2863] and ifInvStackTable, as defined in the
IF-INVERTED-STACK-MIB [RFC2864].

3.1.1 Layering Model

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

An generic EFMCu device can have a number of Physical Coding Sublayer
(PCS) ports, connected to a MAC via 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 comprising an aggregated EFMCu port is represented in the
Interface table as a separate port 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 an actual bitrate
of the active PME or a configured bitrate for pre-activated modems
(note that unassigned PME has its default bitrate).

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 efmCuAvailableStackTable defined in this MIB, specifies

for each PCS a list of PMEs, which can possibly be cross-connected to
that PCS, determined by the cross-connect capability of the device.
This table, modeled after ifStackTable, is read only, reflecting
current cross-connect capability, which can be dynamic in some
implementations (e.g. if PMEs are located on a pluggable module and
the card is pulled out). Note that PME availability per PCS,
described by efmCuAvailableStackTable, 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. efmCuAvailableStackTable, ifStackTable
and efmCuPAFCapacity) SHALL be inspected.

The efmCuAvailableStackTable allows to define new entries

*EdNote: Add efmCuInvAvailableStackTable describing which PCS ports
can be connected to a particular PME. Possibly move these 2 tables to
a separate MIB to make cross-connect capability available to other
MIBs for import (changing the prefix of course from 'efmCu' to say
'if'). *

3.1.2 PME Aggregation Function (PAF)

The PME Aggregation Function (PAF) is OPTIONAL and MAY not be
supported. Note however that the agent is REQUIRED to report on the
PAF capability for all EFMCu ports (2Base-TL and 10Pass-TS).

EdNote: Add more info.

3.1.3 Discovery Operation

This MIB allows a Network Management application to control 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 Discovery mechanism: ifStackTable and
efmCuAvailableStackTable defined. The following pseudo-code defines
an example of Discovery and automatic PME assignment for a generic
PAF enabled multi-PCS EFMCu device, located at Central Office (CO):

```
FOREACH pcs[i] IN co_device
{ // Discover only on PAF enabled ports with room for more PMEs
  IF ( pcs[i].PAFSupported AND pcs[i].NumPMEs < pcs[i].PAFCapacity )
    { dc = pcs[i].DiscoveryCode = MAC[i]; // unique 6 Byte per PCS
      // go over all currently disconnected PMEs, which can
      // pottentially be connected to PCS[i]
      FOREACH pme[j] IN efmCuAvailableStackTable[pcs[i]] AND
                    NOT IN ifInvStackTable[pme[j]] // unassigned
        { pme[j].RemoteDiscoveryCode = dc;      // Set if Clear
          r = pme[j].RemoteDiscoveryCode;       // Get
          IF ( r == dc AND pcs[i].NumPMEs < pcs[i].PAFCapacity)
            { // Remote CPE connected via PME[j] is/was a peer for
              // PCS[i]. Connect this PME to the PCS (there's room)
              ADD pme[j] TO ifStackTable[pcs[i]];
              pcs[i].NumPMEs = pcs[i].NumPMEs + 1;
              // Discover all other currently disconnected PMEs,
              // attached to the same CPE and connect them to the PCS
              // provided there is enough room for more PMEs.
              FOREACH pme[k] IN efmCuAvailableStackTable[pcs[i]] and
                          NOT IN ifInvStackTable[pme[k]]
                { r = pme[k].RemoteDiscoveryCode;      // Get
                  IF ( r == dc AND
                      pcs[i].NumPMEs < pcs[i].PAFCapacity)
                    { ADD pme[k] TO ifStackTable[pcs[i]];
                      pcs[i].NumPMEs = pcs[i].NumPMEs + 1;
                    }
                }
            }
          // Discovered all PMEs which lead to the same CPE and
          // connected them to PCS[i]. Go to the next PCS.
          BREAK;
        }
    }
}
```

The SNMP Agent builds efmCuStackTable 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 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 whith 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 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 completetly 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 required data rate on a particular copper
pair. Sometimes, when the copper line is too long or the noise
environment 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 and a PME port. Setting this object to 'up' instructs a
particular PCS or PME to start 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 efmCuPmeOperStatus defined in this MIB 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
suceed. 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 this MIB 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 this MIB 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 by 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 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                       | Actual 'net' data rate as seen |
|                               | across MII                    |
| 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

3.2 Relation to SHDSL MIB

G.SHDSL.bis modems, similar to PME(s) comprising a 2BaseTL port, are
described in HDSL2-SHDSL-LINE-MIB [I-D.ietf-adslmib-gshdslbis]. Note
that not all attributes of G.SHDSL modems reflected in
HDSL2-SHDSL-LINE-MIB 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 2BaseTL 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 this MIB.

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

3.3 **Relation to VDSL MIB**

   VDSL (DMT) modems, similar to the PME(s) comprising a 10PassTS port,
   are described in VDSL-LINE-EXT-MCM-MIB
   [I-D.ietf-adslmib-vdsl-ext-mcm]. Note that not all attributes of
   VDSL modems reflected in VDSL-LINE-EXT-MCM-MIB 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
   this MIB.

   However, if some functionality, not available in this MIB, 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 MIBs**

   The implementation of EtherLike-MIB [RFC3635] and MAU-MIB [RFC3636]
   is REQUIRED for the EFMCu interfaces. As such EFMCu interfaces
   2Base-TL/10Pass-TS SHALL return an ifType of ethernetCsmacd(6).
   Information on the particular flavor of EFMCu that an interface is
   running is available from ifSpeed in the IF-MIB [RFC2863], and
   ifMauType in the MAU-MIB.

   The MAU-MIB shall be augmented to include the following new values
   for ifMauType (instances of dot2MauType):

   o dot3MauType2BaseTL - voice grade UTP Phy specified in Clause 61
      and 63

   o dot3MauType10PassTS - voice grade UTP Phy specified in Clause 61
      and 62

   EdNote: Should we also include -O/-R subtypes?

4. **MIB Structure**

4.1 **Overview**

   The main management objects defined in this MIB are split into 2
   groups:

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

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

   In addition the efmCuAvailableStackTable is defined at the same
   level.

   The efmCuPme group in turn contains efmCuPme2B and efmCuPme10P
   groups, which define configuration profiles specific to 2BaseTL and
   10PassTS PMEs respectively as well as PME specific status
   information.

4.2 Configuration 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, this
   MIB makes use of configuration profiles, similar to
   HDSL2-SHDSL-LINE-MIB and VDSL-LINE-EXT-MCM-MIB. A profile is a set
   of parameters that can be shared by multiple PME ports using the same
   configuration.

   The configuration profiles are defined in efmCuPme2BConfProfileTable
   and efmCu10PConfProfileTable 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. An
   ability to define new configuration profiles is also provided to
   allow for EFMCu deployment tailored to specific copper environment
   and spectral regulations.

   A specific configuration profile is assigned to a specific PME via
   efmCuPmeAdminProfile object. Alternatively, by assigning a non-empty
   profile to a PCS port via efmCuAdminProfile, all PMEs connected to
   the PCS port are made to share that profile, which comes handy during
   configuration of aggregated ports.

4.3 Mapping of IEEE 802.3ah Managed Objects

   This section contains the mapping between 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 majority of the objects defined in this MIB do not have
   direct counterparts in Clause 30 and instead refer to Clause 45
   registers.

*EdNote: It would be a good idea to update Clause 30 of 802.3ah after
this MIB is approved. I guess this should be done via a maintenence
request. *

```
+-------------------------------+-------------------------------+
| IEEE 802.3 Managed Object     | Corresponding SNMP Object     |
+-------------------------------+-------------------------------+
| oPAF - PME Basic Package      |                               |
| (Mandatory)                   |                               |
| aPAFID                        | ifIndex (IF-MIB)              |
| aPhyEnd                       | efmCuPhySide                  |
| aPHYCurrentStatus             | efmCuStatus                   |
| aPAFSupported                 | efmCuPAFSupported             |
| oPAF - PME Aggregation Package|                               |
| (Optional)                    |                               |
| aPAFAdminState                | efmCuPAFAdminState            |
| aPAFID                        | ifIndex (IF-MIB)              |
| aLocalPAFCapacity             | efmCuPAFCapacity              |
| aLocalPMEAvailable            | efmCuAvailableStackTable      |
| aLocalPMEAggregate            | ifStackTable (IF-MIB)         |
| aRemotePAFSupported           | efmCuRemotePAFSupported       |
| aRemotePAFCapacity            | efmCuRemotePAFCapacity        |
| aRemotePMEAvailable           |                               |
| aRemotePMEAggregate           |                               |
| oPME - 10P/2B Package         |                               |
| (Mandatory)                   |                               |
| aPMEID                        | ifIndex (IF-MIB)              |
| aPMEAdminState                | ifAdminState (IF-MIB)         |
| aPMEStatus                    | efmCuPmeStatus                |
| aPMESNRMgn                    | efmCuPmeSnrMgn                |
| aTCCodingViolations           | efmCuPmeTCCodingErrors        |
| aProfileSelect                | efmCuAdminProfile,            |
|                               | efmCuPmeAdminProfile          |
| aOperatingProfile             | efmCuPmeOperProfile           |
| aPMEFECCorrectedBlocks        | efmCuPme10PFECCorrectedBlocks |
| aPMEFECUncorrectableBlocks    | efmCuPme10PFECUncorrectedBlocks|
+-------------------------------+-------------------------------+
```

                              Table 2

5. 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]
   SnmpAdminString
     FROM SNMP-FRAMEWORK-MIB -- [RFC3411]
   ifIndex, ifSpeed, InterfaceIndex
     FROM IF-MIB              -- [RFC2863]
   MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
     FROM SNMPv2-CONF         -- [RFC2580]
   ;

efmCuMIB MODULE-IDENTITY
   LAST-UPDATED "200407150000Z"  -- July 15, 2004
   ORGANIZATION "IETF Ethernet Interfaces and Hub MIB Working Group"
   CONTACT-INFO
     "WG charter:
       http://www.ietf.org/html.charters/hubmib-charter.html

     Mailing Lists:
       General Discussion: hubmib@ietf.org
       To Subscribe: hubmib-request@ietf.org
       In Body: subscribe your_email_address

     Chair:  Dan Romascanu
     Postal: Avaya
             Atidim Technology Park, Bldg. 3
             Tel Aviv 61131
             Israel
        Tel: +972 3 645 8414
     E-mail: dromasca@avaya.com

     Editor: Edward Beili
     Postal: Actelis Networks Inc.
             25 Bazel St., P.O.B. 10173
             Petach-Tikva 10173
             Israel
             Tel: +972-3-924-3491
     E-mail: 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 Draft P802.3ah/D3.3.

     The following reference is used throughout this MIB module:

     [802.3ah] refers to:
       IEEE Draft P802.3ah/D3.3: 'Draft amendment to -
```

         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 - Media Access Control Parameters, Physical
         Layers and Management Parameters for subscriber access
         networks', 19 April 2003.

      Of particular interest are Clause 61, 'Physical Coding
      Sublayer (PCS) and common specifications, type 10PASS-TS and
      type 2BASE-TL', Clause 30, 'Management', and Clause 45,
      'Management Data Input/Output (MDIO) Interface'.

      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
        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 Internet Society (2004). This version
      of this MIB module is part of RFC XXXX; see the RFC
      itself for full legal notices."

      -- EdNote: Replace XXXX with the actual RFC number &
      -- remove this note

    REVISION   "200407150000Z" -- July 15, 2004
    DESCRIPTION "Initial version, published as RFC XXXX."

    ::= { mib-2 YYY }

      -- EdNote: Replace YYY with a real OID once it is
      -- allocated & remove this note.

   -- 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

      ProfileIndex ::= 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)

      ProfileIndexOrZero ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "d"
        STATUS       current
        DESCRIPTION
          "This textual convention is an extension of the ProfileIndex
          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 which uses this syntax.
          Examples of the usage of zero value migh include situations
          where current operational profile is unknown."
        SYNTAX       Unsigned32 (0..255)

      ProfileIndexList ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "1d:"
        STATUS       current
        DESCRIPTION
          "Represents a list of up to 6 ProfileIndex's.
          The ProfileIndex textual convention defines a greater than
          zero value used to identify a PME profile in the managed EFMCu
          port. The value of this object is a concatenation of zero or
          more (up to 6) octets, where each octet contains an 8-bit
          ProfileIndex value.
          A non-empty profile list specifies a list of alternative
          profiles any of which can be chosen for configuration of an
```

```
      PME. A zero length string value means this list has no members
      (empty). Examples of the usage of empty profile list might
      include situations where configuration profile(s) is unknown,
      or when no common profile(s) need to be referenced and
      sub-interface profile shall take precedence."
   SYNTAX       OCTET STRING (SIZE(0..6))

-- Port Notifications Group

 efmCuPortNotifications OBJECT IDENTIFIER ::= { efmCuPort 0 }

 efmCuLowBandwidth NOTIFICATION-TYPE
   OBJECTS {
     -- ifINdex is not needed here since we are under specific PCS
     ifSpeed,
     efmCuThreshLowBandwidth
   }
   STATUS       current
   DESCRIPTION
     "This notification indicates that EFMCu port's data rate
     reached or droped below a Low Bandwidth Threshold
     (i.e. bandwidth degradation happening in case of PAF when one
     or more PMEs drop)."
      -- EdNote: add throttling limitations here
   ::= { efmCuPortNotifications 1 }

-- PCS Port group

 efmCuPortConfTable OBJECT-TYPE
   SYNTAX       SEQUENCE OF EfmCuPortConfEntry
   MAX-ACCESS  not-accessible
   STATUS       current
   DESCRIPTION
     "Table for Configuration of EFMCu 2BaseTL/10PassTS (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              ProfileIndexList,
          efmCuTargetDataRate            Unsigned32,
          efmCuTargetSnrMgn              Unsigned32,
          efmCuThreshLowBandwidth        Unsigned32,
          efmCuLowBandwidthEnable        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 ignored.

          PAF 'enabled' port with multiple PMEs assigned cannot be
          'disabled'. Attempts to 'disable' such port SHALL be rejected,
          untill 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 ignored 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."
        REFERENCE
          "[802.3ah] 61.2.2, 45.2.3.18.3"
```

             ::= { efmCuPortConfEntry 1 }

         efmCuPAFDiscoveryCode OBJECT-TYPE
           SYNTAX   PhysAddress
           MAX-ACCESS   read-write
           STATUS   current
           DESCRIPTION
             "PAF Discovery Code of the EFMCu port (PCS).
             A unique 6 Byte long code used by the Discovery function, when
             PAF is supported.
             PCS ports incapable of supporting PAF SHALL return a value of
             all zeroes. Attempts to change this object SHALL be ignored in
             this case.
             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 value of this object is read-only for -R port subtypes.
             The initial value of this object for -R ports after reset
             is 0. This value may be changed as a result of writing
             operation on efmCuPAFRemoteDiscoveryCode variable 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 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"
           ::= { efmCuPortConfEntry 2 }

         efmCuAdminProfile OBJECT-TYPE
           SYNTAX   ProfileIndexList
           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 efmCuPme2BConfProfileTable or
             efmCuPme10PConfProfileTable, 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
             configuration profiles. If this list consists of a single

        Profile index, then all PMEs assigned to this EFMCu port SHALL
        be configured according to the Configuration Profile
        referenced by that index. The list, consisting of more than
        one index, allows each PME in the port to be configured
        according to any configuration profile specified in the list.
        An empty list means that each PME in the port is configured
        via corresponding efmCuPmeAdminProfile object.

        This object is writable and readable for the -O subtype
        (2BaseTL-O or 10PassTS-O) EFMCu ports. It is unavailable for
        the -R subtype (2BaseTL-R or 10PassTS-R) ports.

        Note that current operational Profile value is available via
        efmCuPmeOperProfile object.

        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 non-empty 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."
      REFERENCE
        "[802.3ah] 30.11.2.1.6"
      ::= { 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
        64/65B framing overhead (e.g. target data rate of 10Mbps
        shall allow lossless transmission of full-duplex 10Mbps
        Ethernet frame stream with minimal inter-frame gap).

        The value is limited above by 100Mbps 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 should
        be equal to the target data rate or less, if the target data

rate cannot be achieved under spectral restrictions specified
by efmCuAdminProfile/efmCiPmeAdminProfile and with desired SNR
margin. In case the copper environment allows to achieve
higher total data rate than specified by the target, the
excess capability SHALL be converted to additional SNR margin
and spread evenly across all active PMEs assigned to the (PCS)
port.

The value of 999999 means that the target data rate is not
fixed and should be set to the maximum attainable rate during
initialization (Best Effort), under specified spectral
restrictions and with 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
with the error inconsistentValue, if the link is Up or
Initializing.

Note that current Data Rate of the EFMCu port is represented
by ifSpeed object of IF-MIB."
  ::= { 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 initializiation. (The SNR margin
    is the difference between the desired SNR and the actual SNR).

    Note that 802.3ah recommends using default Target SNR Margin
    of 5dB for 2BaseTL ports and 6dB for 10PassTS ports in order
    to achieve 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
    with the error inconsistentValue, if the link is Up or
    Initializing.

        Note that current SNR Margin of the PMEs comprising the EFMCu
        port is represented by efmCuPmeSnrMgn."
      REFERENCE
        "[802.3ah] 61.1.2"
      ::= { efmCuPortConfEntry 5 }

    efmCuThreshLowBandwidth OBJECT-TYPE
      SYNTAX   Unsigned32(0..100000)
      UNITS   "Kbps"
      MAX-ACCESS   read-write
      STATUS   current
      DESCRIPTION
        "This object configures the EFMCu port Low Bandwidth alarm
        threshold. When the current value of ifSpeed for this port
        reaches or drops below this threshold, an
        efmCuLowBandwidth notification MAY be generated if enabled by
        efmCuLowBandwidthEnable.

        The value of 0 means no efmCuLowBandwidth notifications
        SHALL ever be generated.

        This object is read-write for the -O subtype EFMCu ports
        (2BaseTL-O/10PassTS-O) and not available for the -R subtypes."
      ::= { efmCuPortConfEntry 6 }

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

        Value of truth(1) indicates that efmCuLowBandwidth
        notification is enabled. Value of false(0) indicates the it is
        disabled."
      ::= { efmCuPortConfEntry 7 }

    efmCuPortCapabilityTable OBJECT-TYPE
      SYNTAX   SEQUENCE OF EfmCuPortCapabilityEntry
      MAX-ACCESS   not-accessible
      STATUS   current
      DESCRIPTION
        "Table for Capabilities of EFMCu 2BaseTL/10PassTS (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          TruthValue,
    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      TruthValue
  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 or cannot be reached
        because of the link state, SHALL return a value of false(2).

        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 particluar
        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).
        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
          2BaseTL/10PassTS ports. 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),              -- peer Phy cannot be reached
          pmeSubTypeMismatch(1),  -- Assigned PMEs Sub-type Mismatch
          lowBandwidth(2)         -- Low Bandwidth
        }
        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               - peer PHY cannot be reached (e.g.
```

```
                                 no PMEs attached, all PMEs are Down
                                 etc.) More info is available in
                                 efmCuPmeFltStatus.
          pmeSubTypeMismatch  - PMEs in the aggregation group are not
                                 of the same sub-type, e.g. PMEs with
                                 -O and -R subtype
          lowBandwidth        - ifSpeed of the port reached or droped
                                 below efmCuThreshLowBandwidth

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

  efmCuPortSide  OBJECT-TYPE
    SYNTAX        INTEGER {
      subscriber(1),  -- -R sub-type
      office(2),      -- -O sub-type
      unknown(3)      -- no PMEs assigned or PME sub-type mismatch
    }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
      "EFM port mode of operation (subtype).
       The value of 'subscriber' indicates 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
          "Number of PMEs that is currently aggregated by the local PAF
          (assigned to the EFMCu port using 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
          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
          "A 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."
        REFERENCE
          "[802.3ah] 45.2.3.21"
        ::= { efmCuPortStatusEntry 4 }

    efmCuPAFInSmallFragments OBJECT-TYPE
        SYNTAX   Counter32
        MAX-ACCESS   read-only
        STATUS   current
        DESCRIPTION
          "A 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."
        REFERENCE
          "[802.3ah] 45.2.3.22"
        ::= { efmCuPortStatusEntry 5 }

```
      efmCuPAFInLargeFragments OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
          "A 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."
        REFERENCE
          "[802.3ah] 45.2.3.23"
        ::= { efmCuPortStatusEntry 6 }

      efmCuPAFInBadFragments OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
          "A number of fragments which do not fit into the sequence
          expected by the frame assembly function, 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."
        REFERENCE
          "[802.3ah] 45.2.3.25"
        ::= { efmCuPortStatusEntry 7 }

      efmCuPAFInLostFragments OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
          "A 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."
     REFERENCE
       "[802.3ah] 45.2.3.26"
     ::= { efmCuPortStatusEntry 8 }

   efmCuPAFInLostStarts OBJECT-TYPE
     SYNTAX   Counter32
     MAX-ACCESS read-only
     STATUS   current
     DESCRIPTION
       "A 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."
     REFERENCE
       "[802.3ah] 45.2.3.27"
     ::= { efmCuPortStatusEntry 9 }

   efmCuPAFInLostEnds OBJECT-TYPE
     SYNTAX   Counter32
     MAX-ACCESS read-only
     STATUS   current
     DESCRIPTION
       "A 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."
     REFERENCE
       "[802.3ah] 45.2.3.28"
     ::= { efmCuPortStatusEntry 10 }

   efmCuPAFInOverflows OBJECT-TYPE

```
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "A number of fragements, received accross 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."
        REFERENCE
          "[802.3ah] 45.2.3.24"
        ::= { efmCuPortStatusEntry 11 }

    -- PME Notifications Group

     efmCuPmeNotifications OBJECT IDENTIFIER ::= { efmCuPme 0 }

     -- EdNote: Should I add more notifications here, for example
     --        efmCuPmePerfES,
     --        efmCuPmePerfSES,
     --        efmCuPmePerfCRCanomalies,
     --        efmCuPmePerfLOSWS,
     --        efmCuPmePerfUAS,
     --        efmCuPmeDeviceFault,
     --        efmCuPmeLocalPowerLoss
     -- Another option would be to reference HDSL2-SHDSL-LINE-MIB
     -- or VDSL-LINE-EXT-MCM-MIB, reusing notifications defined there.

     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."
         -- EdNote: add throttling limitations here
       ::= { 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."
        -- EdNote: add throttling limitations here
      ::= { 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."
        -- EdNote: add throttling limitations here
      ::= { 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 requested
        configuration profile."
        -- EdNote: add throttling limitations here
      ::= { efmCuPmeNotifications 4 }

    efmCuPmeProtocolInitFailure NOTIFICATION-TYPE
      OBJECTS {
        efmCuPmeFltStatus,
        efmCuPmeOperSubType
      }
      STATUS      current
      DESCRIPTION
        "This notification indicates that peer PME was using
        incompatible protocol during initialization."
        -- EdNote: add throttling limitations here
      ::= { 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
    2BaseTL/10PassTS PME ports (modems). Configuration of aspects
    specific to 2BaseTL or 10PassTS 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          ProfileIndex,
    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) sub-type 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. Actual value will be
                                      set by -O link partner during
                                      initialization (handshake).
              ieee2BaseTLor10PassTSO - PME shall operate as 2BaseTL-O
                                      (preferred) or 10PassTS-O. Actual
                                      value will be set during
                                      initialization depending on -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. Actual
                                      value will be set during
                                      initialization depending on -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 ignored if the link
          is Up or Initializing.
          Attempts to change this object to an unsupported subtype
          (see efmCuPmeSubTypesSupported) SHALL be rejected.

          The current operational sub type is indicated by
          efmCuPmeOperSubType variable.

          If a Clause 45 MDIO Interface to the PMA/PMD is present, then
          this object combines values of the Port sub-type 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   ProfileIndex
                 MAX-ACCESS   read-write
                 STATUS   current
                 DESCRIPTION
                   "Desired PME configuration profile. This object is a pointer
                   to an entry in either efmCuPme2BConfProfileTable or
                   efmCuPme10PConfProfileTable, depending on the current
                   operating SubType of the PME. The value of this object is the
                   index of the referenced profile.
                   Note that if PME is assigned to a PCS port and value of
                   efmCuAdminProfile for that port is not an empty list, then the
                   profile(s) referenced by efmCuAdminProfile takes precedence
                   over the profile referenced by efmCuPmeAdminProfile, i.e. the
                   PME SHALL be configured according to efmCuAdminProfile.

                   This object is writable and readable for the CO subtype PMEs
                   (2BaseTL-O or 10PassTS-O). It is unavailable for the CPE
                   subtype (2BaseTL-R or 10PassTS-R).

                   Note that current operational Profile value is available via
                   efmCuPmeOperProfile object.

                   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."
                 REFERENCE
                   "[802.3ah] 30.11.2.1.6"
                 ::= { efmCuPmeConfEntry 2 }

             efmCuPAFRemoteDiscoveryCode OBJECT-TYPE
                 SYNTAX   PhysAddress
                 MAX-ACCESS   read-write
                 STATUS   current
                 DESCRIPTION
                   "PAF Remote Discovery Code of the PME port at CO.
                   A 6 Byte long Discovery Code of the peer PCS connected via
                   the PME.
                   Reading this object results in a Discovery Get operation.
                   Writing a zero to this object 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.
            This object does not exist in CPE port subtypes. A zero length
            octet string SHALL be returned for CPE port subtypes and also
            when PAF aggregation is not enabled.

            Discovery MUST be performed when the link is Down.
            Attempts to change this object MUST be rejected 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, a 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 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 will map 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, a efmCuPmeSnrMgnCrossing
             notification MAY be generated, if enabled by
             efmCuPmeSnrMgnCrossingEnable.

             This object is writable for the CO subtype PMEs
             (2BaseTL-O/10PassTS-R). 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 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 will map 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.

             Value of truth(1) indicates that efmCuPmeLineAtnCrossing
             notification is enabled. Value of false(0) indicates the it 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.

       Value of truth(1) indicates that efmCuPmeSnrMgnCrossing
       notification is enabled. Value of false(0) indicates the it 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.

       Value of truth(1) indicates that efmCuPmeDeviceFault
       notification is enabled. Value of false(0) indicates the it 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.

       Value of truth(1) indicates that efmCuPmeConfigInitFailure
       notification is enabled. Value of false(0) indicates the it 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.

       Value of truth(1) indicates that efmCuPmeProtocolInitFailure
       notification is enabled. Value of false(0) indicates the it is
       disabled."
     ::= { efmCuPmeConfEntry 10 }
```

```
        efmCuPmeCapabilityTable OBJECT-TYPE
          SYNTAX       SEQUENCE OF EfmCuPmeCapabilityEntry
          MAX-ACCESS  not-accessible
          STATUS       current
          DESCRIPTION
            "Table for Configuration of common aspects for EFMCu
            2BaseTL/10PassTS PME ports (modems). Configuration of aspects
            specific to 2BaseTL or 10PassTS PME types is represented in
            efmCuPme2BConfTable and 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 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 }
          ::= { 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 sub-types. This is a bitmap of possible
            sub-types. 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
```

        An actual mode of operation is determined by
        efmCuPmeAdminSubType.

        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
        "Table for Configuration of common aspects for EFMCu
        2BaseTL/10PassTS PME ports (modems). Configuration of aspects
        specific to 2BaseTL or 10PassTS PME types is represented in
        efmCuPme2BConfTable and efmCuPme10PConfTable respectively.

        Entries in this table MUST be maintained in a persistent
        manner."
      ::= { efmCuPme 3 }

    efmCuPmeStatusEntry OBJECT-TYPE
      SYNTAX   EfmCuPmeStatusEntry
      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 }
      ::= { efmCuPmeStatusTable 1 }

    EfmCuPmeStatusEntry ::=
      SEQUENCE {
        efmCuPmeOperStatus   INTEGER,
        efmCuPmeFltStatus   BITS,
        efmCuPmeOperSubType   INTEGER,
        efmCuPmeOperProfile   ProfileIndexOrZero,
        efmCuPmeSnrMgn   Integer32,

```
        efmCuPmePeerSnrMgn              Integer32,
        efmCuPmeLineAtn                Integer32,
        efmCuPmePeerLineAtn            Integer32,
        efmCuPmeTCCodingErrors        Counter32
      }

efmCuPmeOperStatus  OBJECT-TYPE
  SYNTAX       INTEGER {
    up(1),                    -- link is Up
    downNotReady(2),          -- link is Down and not Ready
    downReady(3),             -- link is Down and Ready
    init(4)                   -- link is Initializing
  }
  MAX-ACCESS  read-only
  STATUS      current
  DESCRIPTION
    "Current PME link Operational Status. Possible values are:
      up(1)            - link is Up and ready to pass 64/65B
                           encoded frames or fragments.
      downNotReady(2)  - 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)     - link is Down and the PME detects Handshake
                           tones from its peer.
      init(4)          - link is initializing, as a result of
                           ifAdminStatus being set to 'up' for a
                           particular PME or a PCS the PME is
                           connected to.

    This object is intended to supplement Down 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),      -- Loss of Framing
    snrMgnDefect(1),       -- SNR Margin dropped below Threshold
    lineAtnDefect(2),      -- Line Attenuation exceeds Threshold
```

```
      deviceFault(3),          -- Vendor-dependent diag or self-test
                               -- fault
      configInitFailure(4),   -- Configuration Init. failure
      protocolInitFailure(5) -- Protocol Initialization failure
    }
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
  "Current 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 64/65B 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 configuration profile,
                          requested during initialization.
    protocolInitFailure - Protocol initialization failure,
                          due to incompatible protocol used by
                          the Peer PME during init (that could
                          happen if a peer PMD is G.SDHSL/VDSL
                          modem for 2BaseTL/10PassTS PME
                          respectively).

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

  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 sub-type 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 operational sub type of the PME can be configured via
           efmCuPmeAdminSubType variable.

           If a Clause 45 MDIO Interface to the PMA/PMD is present, then
           this object combines values of the Port sub-type 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       ProfileIndexOrZero
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
           "PME current operating Profile. This object is a pointers to
           an entry in either efmCuPme2BConfProfileTable or
           efmCuPme10PConfProfileTable, depending on the current
           operating SubType of the PME as indicated by
           efmCuPmeOperSubType.
           The value of zero indicates that 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)
```

```
              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.

                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)
              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.

                This object is not supported by -R PME subtypes.

                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)
              UNITS        "dB"
              MAX-ACCESS   read-only
              STATUS       current
              DESCRIPTION
                "The current Line Attenuation in dB as perceived by the local
                PME.

                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)
  UNITS        "dB"
  MAX-ACCESS   read-only
  STATUS       current
  DESCRIPTION
    "The current Line Attenuation in dB as perceived by the remote
    (link partner) PME.

    This object is not supported by CPE port subtypes.

    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 }

efmCuPmeTCCodingErrors OBJECT-TYPE
  SYNTAX       Counter32
  MAX-ACCESS   read-only
  STATUS       current
  DESCRIPTION
    "A 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.

    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)."
  REFERENCE
    "[802.3ah] 61.3.3.1, 45.2.6.12"
  ::= { efmCuPmeStatusEntry 9 }

-- 2BaseTL specific PME group

efmCuPme2B       OBJECT IDENTIFIER ::= { efmCuPme 5 }

efmCuPme2BConfProfileTable OBJECT-TYPE
  SYNTAX       SEQUENCE OF EfmCuPme2BConfProfileEntry
  MAX-ACCESS   not-accessible
  STATUS       current
  DESCRIPTION
    "This table supports definitions of configuration profiles for
    2BaseTL PMEs.
    First 12 entries in this table SHALL always be defined as
    follows (see 802.3ah Annex 63A):
    -------+------+-----+--------+------------------
```

```
     Profile  Rate  Power  Region   Constellation
      index  (Kbps) (dBm) (G.991.2)
     -------+------+-----+--------+------------------
        1      5696  13.5    Annex A  32-TCPAM
        2      3072  13.5    Annex A  32-TCPAM
        3      2048  13.5    Annex A  16-TCPAM
        4      1024  13.5    Annex A  16-TCPAM
        5       704  13.5    Annex A  16-TCPAM
        6       512  13.5    Annex A  16-TCPAM
        7      5696  14.5    Annex B  32-TCPAM
        8      3072  14.5    Annex B  32-TCPAM
        9      2048  14.5    Annex B  16-TCPAM
       10      1024  13.5    Annex B  16-TCPAM
       11       704  13.5    Annex B  16-TCPAM
       12       512  13.5    Annex B  16-TCPAM
```

    These default entries SHALL be created by default and MUST
    not be deleted.

    Entries following the first 12, can be dynamically created and
    deleted, to provide custom profiles.

    This table MUST be maintained in a persistent manner."
  REFERENCE
    "[802.3ah] Annex 63A, 30.11.2.1.6"
  ::= { efmCuPme2B 2 }

efmCuPme2BConfProfileEntry OBJECT-TYPE
  SYNTAX   EfmCuPme2BConfProfileEntry
  MAX-ACCESS not-accessible
  STATUS   current
  DESCRIPTION
    "Each entry corresponds to a single 2BaseTL PME configuration
    profile. Each profile contains a set of configuration
    parameters, which are applied to all 2BaseTL-O PMEs
    assigned to the 2BaseTL (PCS) port, referencing that profile
    via efmCuAdminProfile object or, if efmCuAdminProfile is zero,
    all 2BaseTL-O PMEs, referencing that profile via
    efmCuPmeAdminProfile.

    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 (first 12) MUST not be removed."
  INDEX { efmCuPme2BProfileIndex }
  ::= { efmCuPme2BConfProfileTable 1 }

```
        EfmCuPme2BConfProfileEntry ::=
          SEQUENCE {
            efmCuPme2BProfileIndex          ProfileIndex,
            efmCuPme2BProfileDescr          SnmpAdminString,
            efmCuPme2BRegion                INTEGER,
            efmCuPme2BDataRate              Unsigned32,
            efmCuPme2BPower                 Unsigned32,
            efmCuPme2BConstellation         INTEGER,
            efmCuPme2BProfileRowStatus      RowStatus
          }

        efmCuPme2BProfileIndex OBJECT-TYPE
          SYNTAX      ProfileIndex
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
            "2BaseTL PME Profile index.
            This object is the unique index associated with this profile.
            Entries in this table are referenced via efmCuAdminProfile
            or efmCuPmeAdminProfile objects."
          ::= { efmCuPme2BConfProfileEntry 1 }

        efmCuPme2BProfileDescr OBJECT-TYPE
          SYNTAX      SnmpAdminString (SIZE(0..255))
          MAX-ACCESS  read-create
          STATUS      current
          DESCRIPTION
            "A textual string containing information about 2BaseTL PME
            Profile. The string MAY include information about data rate
            and spectral limitations of this particular profile."
          ::= { efmCuPme2BConfProfileEntry 2 }

        efmCuPme2BRegion  OBJECT-TYPE
          SYNTAX      INTEGER {
            regionA(1),  -- Annex A
            regionB(2),  -- Annex B
            regionC(3)   -- Annex C
          }
          MAX-ACCESS  read-create
          STATUS      current
          DESCRIPTION
            "Desired Power Spectral Density (PSD) Regional setting as
            specified in Regional Annex of [ITU-T G.991.2] to operate
            under.

            Possible values for this object are:
              regionA      -- Annex A
              regionB      -- Annex B
```

       regionC   -- Annex C

    Changing Regional Annex must be performed when the link is
    Down. Attempts to change this object MUST be rejected 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 Region bits in the 2B general
    parameter register."
  REFERENCE
    "[802.3ah] 45.2.1.42"
  ::= { efmCuPme2BConfProfileEntry 3 }

efmCuPme2BDataRate OBJECT-TYPE
  SYNTAX Unsigned32(0..5696)
  UNITS   "Kbps"
  MAX-ACCESS read-create
  STATUS   current
  DESCRIPTION
    "Desired 2BaseTL PME Data Rate.
    The rate is fixed when the value of this object is n x 64Kbps,
    where n=3..60 for 16-TCPAM and n=12..89 for 32-TCPAM.
    The value of 0 means that data rate is not fixed but is
    adaptive and should be set to the maximum attainable rate
    during line probing.

    If a Clause 45 MDIO Interface to the PME is present, then this
    object maps to the Min/Max Data Rate1 bits in the 2B PMD
    parameters register."
  REFERENCE
    "[802.3ah] 45.2.1.43"
  ::= { efmCuPme2BConfProfileEntry 4 }

efmCuPme2BPower OBJECT-TYPE
  SYNTAX   Unsigned32(10..42)
  UNITS   "0.5 dBm"
  MAX-ACCESS read-create
  STATUS   current
  DESCRIPTION
    "Desired Signal Transmit Power. Multiple of 0.5dBm.

    Changing of the Signal Transmit Power must be performed when
    the link is Down. Attempts to change this object MUST be
    rejected 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 will map to the Power1 bits in the 2B PMD
          parameters register"
        REFERENCE
          "[802.3ah] 45.2.1.43"
        ::= { efmCuPme2BConfProfileEntry 5 }

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

          If a Clause 45 MDIO Interface to the PME is present, then this
          object map to the Constellation1 bits in the 2B general
          parameter register."
        REFERENCE
           "[802.3ah] 45.2.1.43"
        ::= { efmCuPme2BConfProfileEntry 6 }

     efmCuPme2BProfileRowStatus OBJECT-TYPE
        SYNTAX   RowStatus
        MAX-ACCESS read-create
        STATUS   current
        DESCRIPTION
          "This object controls creation/deletion of the associated
          entry in efmCuPme2BConfProfileTable 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."
        ::= { efmCuPme2BConfProfileEntry 7 }

     -- 10PassTS specific PME group

     efmCuPme10P   OBJECT IDENTIFIER ::= { efmCuPme 6 }

     efmCuPme10PConfProfileTable OBJECT-TYPE
        SYNTAX   SEQUENCE OF EfmCuPme10PConfProfileEntry
        MAX-ACCESS not-accessible
        STATUS   current
        DESCRIPTION

         "This table supports definitions of configuration profiles for
         10PassTL PMEs.
         First 22 entries in this table SHALL always be defined as
         follows (see 802.3ah Annex 62B.3):
         -------+--------+----+---------+-----+------------
         Profile Bandplan UPBO BandNotch DRate URate
          Index PSDMask# p#   p#   p#   p#
         -------+--------+----+---------+-----+------------

Profile Index	Bandplan PSDMask#	UPBO p#	BandNotch p#	DRate p#	URate p#
1	1	3	2,6,10,11	20	20(default)
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 by default and MUST
         not be deleted.

         Entries following the first 22, can be dynamically created and
         deleted, to provide custom profiles.

         This table MUST be maintained in a persistent manner."
      REFERENCE
         "[802.3ah] Annex 62B.3, 30.11.2.1.6"
      ::= { efmCuPme10P 1 }

    efmCuPme10PConfProfileEntry OBJECT-TYPE
      SYNTAX   EfmCuPme10PConfProfileEntry
      MAX-ACCESS not-accessible
      STATUS   current
      DESCRIPTION
         "Each entry corresponds to a single 10PassTS PME configuration

```
      profile. Each profile contains a set of configuration
      parameters, which are applied to all PMEs assigned to the
      10PassTS (PCS) port, referencing that profile via
      efmCuAdminProfile object, or, if efmCuAdminProfile is zero,
      to all PMEs referencing that profile via efmCuPmeAdminProfile.
      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 (first 22) MUST not be removed."
    INDEX { efmCuPme10PProfileIndex }
    ::= { efmCuPme10PConfProfileTable 1 }

EfmCuPme10PConfProfileEntry ::=
  SEQUENCE {
    efmCuPme10PProfileIndex          ProfileIndex,
    efmCuPme10PProfileDescr          SnmpAdminString,
    efmCuPme10PBandplanPSDMskProfile  INTEGER,
    efmCuPme10PUPBOReferenceProfile   INTEGER,
    efmCuPme10PBandNotchProfiles     BITS,
    efmCuPme10PPayloadURateProfile    INTEGER,
    efmCuPme10PPayloadDRateProfile    INTEGER,
    efmCuPme10PProfileRowStatus       RowStatus
  }

efmCuPme10PProfileIndex OBJECT-TYPE
  SYNTAX      ProfileIndex
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "10PassTS PME Profile Index.
    This object is the unique index associated with this profile.
    Entries in this table are referenced via efmCuAdminProfile or
    efmCuPmeAdminProfile."
  ::= { efmCuPme10PConfProfileEntry 1 }

efmCuPme10PProfileDescr OBJECT-TYPE
  SYNTAX      SnmpAdminString (SIZE(0..255))
  MAX-ACCESS  read-create
  STATUS      current
  DESCRIPTION
    "A textual string containing information about 10PassTS PME
    Profile. The string MAY include information about data rate
    and spectral limitations of this particular profile."
  ::= { efmCuPme10PConfProfileEntry 2 }

efmCuPme10PBandplanPSDMskProfile  OBJECT-TYPE
  SYNTAX   INTEGER {-- PSD Mask                Bands      Bandplan
```

```
          profile1(1),    -- T1.424/T-U P1 FTTCab.M1  x/D/U/D/U  A
          profile2(2),    -- T1.424/T-U P1 FTTEx.M1
          profile3(3),    -- T1.424/T-U P1 FTTCab.M2
          profile4(4),    -- T1.424/T-U P1 FTTEx.M2
          profile5(5),    -- T1.424/T-U P1 FTTCab.M1  D/D/U/D/U
          profile6(6),    -- T1.424/T-U P1 FTTEx.M1
          profile7(7),    -- T1.424/T-U P1 FTTCab.M2
          profile8(8),    -- T1.424/T-U P1 FTTEx.M2
          profile9(9),    -- T1.424/T-U P1 FTTCab.M1  U/D/U/D/x
          profile10(10), -- T1.424/T-U P1 FTTEx.M1
          profile11(11), -- T1.424/T-U P1 FTTCab.M2
          profile12(12), -- T1.424/T-U P1 FTTEx.M2
          profile13(13), -- TS1 101 270-1 Pcab.M1.A  x/D/U/D/U  B
          profile14(14), -- TS1 101 270-1 Pcab.M1.B
          profile15(15), -- TS1 101 270-1 Pex.P1.M1
          profile16(16), -- TS1 101 270-1 Pex.P2.M1
          profile17(17), -- TS1 101 270-1 Pcab.M2
          profile18(18), -- TS1 101 270-1 Pex.P1.M2
          profile19(19), -- TS1 101 270-1 Pex.P2.M2
          profile20(20), -- TS1 101 270-1 Pcab.M1.A  U/D/U/D/x
          profile21(21), -- TS1 101 270-1 Pcab.M1.B
          profile22(22), -- TS1 101 270-1 Pex.P1.M1
          profile23(23), -- TS1 101 270-1 Pex.P2.M1
          profile24(24), -- TS1 101 270-1 Pcab.M2
          profile25(25), -- TS1 101 270-1 Pex.P1.M2
          profile26(26), -- TS1 101 270-1 Pex.P2.M2
          profile27(27), -- G.993.1 F.1.2.1 (VDSLoPOTS) x/D/U/D/U  F
          profile28(28), -- G.993.1 F.1.2.2 (VDSLoTCM-ISDN)
          profile29(29)  -- G.993.1 F.1.2.3 (PSD reduction)
         }
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
           "10PassTS PME Bandplan and PSD Mask profile,
           as specified in 802.3ah Annex 62A. Possible values are:
           --------------+-----------------------+-----------+---------
           Profile Name   PSD Mask                    Bands     Bandplan
           --------------+-----------------------+-----------+---------
           profile1(1)   - T1.424/T-U P1 FTTCab.M1  x/D/U/D/U  A
           profile2(2)   - T1.424/T-U P1 FTTEx.M1
           profile3(3)   - T1.424/T-U P1 FTTCab.M2
           profile4(4)   - T1.424/T-U P1 FTTEx.M2
           profile5(5)   - T1.424/T-U P1 FTTCab.M1  D/D/U/D/U
           profile6(6)   - T1.424/T-U P1 FTTEx.M1
           profile7(7)   - T1.424/T-U P1 FTTCab.M2
           profile8(8)   - T1.424/T-U P1 FTTEx.M2
           profile9(9)   - T1.424/T-U P1 FTTCab.M1  U/D/U/D/x
           profile10(10) - T1.424/T-U P1 FTTEx.M1
```

```
        profile11(11) - T1.424/T-U P1 FTTCab.M2
        profile12(12) - T1.424/T-U P1 FTTEx.M2
        profile13(13) - TS1 101 270-1 Pcab.M1.A  x/D/U/D/U  B
        profile14(14) - TS1 101 270-1 Pcab.M1.B
        profile15(15) - TS1 101 270-1 Pex.P1.M1
        profile16(16) - TS1 101 270-1 Pex.P2.M1
        profile17(17) - TS1 101 270-1 Pcab.M2
        profile18(18) - TS1 101 270-1 Pex.P1.M2
        profile19(19) - TS1 101 270-1 Pex.P2.M2
        profile20(20) - TS1 101 270-1 Pcab.M1.A  U/D/U/D/x
        profile21(21) - TS1 101 270-1 Pcab.M1.B
        profile22(22) - TS1 101 270-1 Pex.P1.M1
        profile23(23) - TS1 101 270-1 Pex.P2.M1
        profile24(24) - TS1 101 270-1 Pcab.M2
        profile25(25) - TS1 101 270-1 Pex.P1.M2
        profile26(26) - TS1 101 270-1 Pex.P2.M2
        profile27(27) - G.993.1 F.1.2.1 (VDSLoPOTS) x/D/U/D/U  F
        profile28(28) - G.993.1 F.1.2.2 (VDSLoTCM-ISDN)
        profile29(29) - G.993.1 F.1.2.3 (PSD reduction)

        This object maps to the aBandplanPSDMaskProfile attribute
        in Clause 30."
      REFERENCE
        "[802.3ah] Annex 62A, 30.5.1.1.22"
      ::= { efmCuPme10PConfProfileEntry 3 }

    efmCuPme10PUPBOReferenceProfile  OBJECT-TYPE
      SYNTAX   INTEGER {-- Reference PSD
        profile1(1),   -- T1.424/T-U         Noise A M1
        profile2(2),   -- T1.424/T-U         Noise A M2
        profile3(3),   -- T1.424/T-U         Noise F M1
        profile4(4),   -- T1.424/T-U         Noise F M2
        profile5(5),   -- ETSI TS 101 270-1  Noise A&B
        profile6(6),   -- ETSI TS 101 270-1  Noise C
        profile7(7),   -- ETSI TS 101 270-1  Noise D
        profile8(8),   -- ETSI TS 101 270-1  Noise E
        profile9(9)    -- ETSI TS 101 270-1  Noise F
      }
      MAX-ACCESS  read-create
      STATUS      current
      DESCRIPTION
        "10PassTS PME Upstream Power Back-Off (UPBO) Reference PSD
        Profile, as specified in 802.3ah Annex 62A. Possible values
        are:
          profile1(1)   - T1.424/T-U         Noise A M1
          profile2(2)   - T1.424/T-U         Noise A M2
          profile3(3)   - T1.424/T-U         Noise F M1
          profile4(4)   - T1.424/T-U         Noise F M2
```

```
         profile5(5)    - ETSI TS 101 270-1  Noise A&B
         profile6(6)    - ETSI TS 101 270-1  Noise C
         profile7(7)    - ETSI TS 101 270-1  Noise D
         profile8(8)    - ETSI TS 101 270-1  Noise E
         profile9(9)    - ETSI TS 101 270-1  Noise F


      This object maps to the aUPBOReferenceProfile attribute
      in Clause 30."
    REFERENCE
      "[802.3ah] Annex 62A.3.4, 30.5.1.1.23"
    ::= { efmCuPme10PConfProfileEntry 4 }

  efmCuPme10PBandNotchProfiles   OBJECT-TYPE
    SYNTAX  BITS {  -- G.991.3  T1.424/T-U TS101 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
    }
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION
      "10PassTS PME Egress Control Band Notch Profile bitmap,
      as specified in 802.3ah Annex 62A. Possible values are:
      --------------+---------+----------+-----------+------+-----
      Profile Name    G.991.3  T1.424/T-U TS101 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 value of 0x0622 selects
     profiles 2,6,10 and 11.

     This object maps to the aBandNotchProfile attribute
     in Clause 30."
   REFERENCE
     "[802.3ah] Annex 62A.3.5, 30.5.1.1.19"
   ::= { efmCuPme10PConfProfileEntry 5 }

efmCuPme10PPayloadURateProfile OBJECT-TYPE
   SYNTAX   INTEGER {-- Upstream Payload Rate (Mbps)
     profile5(5),   -- 2.5
     profile10(10),   -- 5
     profile15(15),   -- 7.5
     profile20(20),   -- 10
     profile25(25),   -- 12.5
     profile30(30),   -- 15
     profile50(50),   -- 25
     profile70(70),   -- 35
     profile100(100)   -- 50
   }
   MAX-ACCESS read-create
   STATUS   current
   DESCRIPTION
     "10PassTS PME Upstream Payload Rate Profile,
     as specified in 802.3ah 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.

     This object maps to the aPayloadRateProfileUpstream
     attribute in Clause 30."
   REFERENCE
```

          "[802.3ah] Annex 62A.3.6, 30.5.1.1.20"
        ::= { efmCuPme10PConfProfileEntry 6 }

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

         This object maps to the aPayloadRateProfileDownstream
         attribute in Clause 30."
       REFERENCE
         "[802.3ah] Annex 62A.3.6, 30.5.1.1.21"
        ::= { efmCuPme10PConfProfileEntry 7 }

     efmCuPme10PPProfileRowStatus OBJECT-TYPE
       SYNTAX       RowStatus

```
          MAX-ACCESS  read-create
          STATUS      current
          DESCRIPTION
            "This object controls creation/deletion of the associated
            entry in efmCuPme10PConfProfileTable 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."
          ::= { efmCuPme10PConfProfileEntry 8 }


     efmCuPme10PStatusTable OBJECT-TYPE
          SYNTAX      SEQUENCE OF EfmCuPme10PStatusEntry
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
            "Table reflecting status of EFMCu 10PassTS PMEs (modems)."
          ::= { efmCuPme10P 2 }

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

     EfmCuPme10PStatusEntry ::=
          SEQUENCE {
            efmCuPme10PElectricalLength       Integer32,
            efmCuPme10PFECCorrectedBlocks     Counter32,
            efmCuPme10PFECUncorrectedBlocks   Counter32
            -- EdNote: To be continued
          }

     efmCuPme10PElectricalLength  OBJECT-TYPE
          SYNTAX      Integer32(0..8192|65535)
          UNITS       "m"
          MAX-ACCESS  read-only
          STATUS      current
          DESCRIPTION
            "Electrical Length in meters as perceived by the 10PassTS PME
            after the link is established.
            The value of 65535 is returned if the link is Down or
            Initializing or the PME is unable to estimate the Electrical
            Length.
```

           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"
        ::= { efmCuPme10PStatusEntry 1 }

     efmCuPme10PFECCorrectedBlocks OBJECT-TYPE
        SYNTAX   Counter32
        MAX-ACCESS read-only
        STATUS   current
        DESCRIPTION
           "A count of received and corrected FEC codewords in 10Pass-TS
           PME.

           This object maps to 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"
        REFERENCE
           "[802.3ah] 45.2.1.22"
        ::= { efmCuPme10PStatusEntry 2 }

     efmCuPme10PFECUncorrectedBlocks OBJECT-TYPE
        SYNTAX   Counter32
        MAX-ACCESS read-only
        STATUS   current
        DESCRIPTION
           "A count of received FEC codewords in 10Pass-TS PME, which are
           uncorrectable.

           This object maps to 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"
        REFERENCE
           "[802.3ah] 45.2.1.23"
        ::= { efmCuPme10PStatusEntry 3 }

     -- efmCuAvailableStackTable for use in Discovery

     efmCuAvailableStackTable OBJECT-TYPE
        SYNTAX   SEQUENCE OF EfmCuAvailableStackEntry
        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 in
         ifStackTable). In particular, it contains information on
         which PCS ports (sublayers) can possibly run 'on top of'
         which PMEs (sublayers), as determined by cross-connect
         capability of the EFMCu device, where each sub-layer
         corresponds to a conceptual row in the ifTable. For example,
         when the PCS port with ifIndex value x can be connected
         to run on top of the PME with ifIndex value y, then this table
         contains:

           efmCuAvailableStackStatus.x.y=capable

         Note that there's always at least on PCS for each PME and at
         least one PME for each PCS in the EFMCu devices, with
         efmCuPAFCapacity and efmCuPeerPAFCapacity indicating
         maximum number of PMEs which can be aggregated by local and
         remote PCS port respectively.
         This table is read only as it describes device capability"
       REFERENCE
         "ifStackTable of RFC 2863"
       ::= { efmCuObjects 3 }

     efmCuAvailableStackEntry OBJECT-TYPE
       SYNTAX   EfmCuAvailableStackEntry
       MAX-ACCESS not-accessible
       STATUS   current
       DESCRIPTION
         "Information on a particular relationship between two sub-
         layers, specifying that one sub-layer (PCS) runs on 'top' of
         the other sub-layer (PME). Each sub-layer corresponds to a
         conceptual row in the ifTable (interface index for PCS and
         PME respectively)."
       INDEX {
         efmCuAvailableStackHigherLayer,
         efmCuAvailableStackLowerLayer
       }
       ::= { efmCuAvailableStackTable 1 }

     EfmCuAvailableStackEntry ::=
        SEQUENCE {
          efmCuAvailableStackHigherLayer InterfaceIndex,
          efmCuAvailableStackLowerLayer   InterfaceIndex,
          efmCuAvailableStackStatus   INTEGER
        }

```
      efmCuAvailableStackHigherLayer  OBJECT-TYPE
        SYNTAX       InterfaceIndex
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
          "The value of ifIndex corresponding to the higher sub-layer
          of the 'cross-connect capability' relationship, i.e., the
          PCS sub-layer which MAY run on 'top' of the PME sub-layer
          identified by the corresponding instance of
          efmCuAvailableStackLowerLayer."
        ::= { efmCuAvailableStackEntry 1 }


      efmCuAvailableStackLowerLayer  OBJECT-TYPE
        SYNTAX       InterfaceIndex
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
          "The value of ifIndex corresponding to the lower sub-layer
          of the 'cross-connect capability' relationship, i.e., the
          PME sub-layer which MAY run 'below' the PCS sub-layer
          identified by the corresponding instance of
          efmCuAvailableStackHigherLayer."
        ::= { efmCuAvailableStackEntry 2 }

      efmCuAvailableStackStatus  OBJECT-TYPE
        SYNTAX       INTEGER {
          capable(1),
          outOfService(2)
        }
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
          "The status of the 'cross-connect capability' relationship
          between two sub-layers. The following values are defined:
            capable(1)      - the PME sub-layer interface, identified by
                              the efmCuAvailableStackLowerLayer MAY be
                              connected to run 'below' the PCS sub-layer
                              interface, identified by the
                              efmCuAvailableStackLowerLayer.
            outOfService(2) - the PME sub-layer interface cannot be
                              connected due to unavailability of the
                              interface.

          Note that PME availability per PCS, indicated by 'capable'
          value, 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. efmCuAvailableStackTable,
         ifStackTable and efmCuPAFCapacity) SHALL be inspected.

         This object is read only, unlike ifStackStatus, as it
         describes the device capability."
       ::= { efmCuAvailableStackEntry 3 }

      --
      -- Conformance Statements
      --

       efmCuGroups   OBJECT IDENTIFIER ::= { efmCuConformance 1 }

       efmCuCompliances OBJECT IDENTIFIER ::= { efmCuConformance 2 }

       -- Object Groups

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

       efmCuPAFGroup OBJECT-GROUP
         OBJECTS {
           efmCuPeerPAFSupported,
           efmCuPAFCapacity,
           efmCuPeerPAFCapacity,
           efmCuPAFAdminState,
           efmCuPAFDiscoveryCode,
           efmCuPAFRemoteDiscoveryCode,
           efmCuNumPMEs,
           efmCuAvailableStackStatus
         }
         STATUS   current
         DESCRIPTION
           "A collection of objects required for 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,
          efmCuPmeTCCodingErrors,
          efmCuPmeThreshLineAtn,
          efmCuPmeThreshSnrMgn
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects providing information about
          a 2BaseTL/10PassTS PME."
        ::= { efmCuGroups 4 }

      efmCuAlarmConfGroup OBJECT-GROUP
        OBJECTS {
          efmCuThreshLowBandwidth,
          efmCuLowBandwidthEnable,
          efmCuPmeThreshLineAtn,
```

```
           efmCuPmeLineAtnCrossingEnable,
           efmCuPmeThreshSnrMgn,
           efmCuPmeSnrMgnCrossingEnable,
           efmCuPmeLineAtnCrossingEnable,
           efmCuPmeDeviceFaultEnable,
           efmCuPmeConfigInitFailEnable,
           efmCuPmeProtocolInitFailEnable
   --        efmCuPmeThreshES,
   --        efmCuPmethreshSES,
   --        efmCuPmeThreshCRCanomalies,
   --        efmCuPmeThreshLOSWS,
   --        efmCuPmeThreshUAS
          }
          STATUS      current
          DESCRIPTION
            "A collection of objects required for configuration of alarm
            thresholds and notifications in EFMCu ports."
          ::= { efmCuGroups 5 }

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

     efmCuPme2BProfileGroup OBJECT-GROUP
          OBJECTS {
            efmCuPme2BProfileDescr,
            efmCuPme2BRegion,
            efmCuPme2BDataRate,
            efmCuPme2BPower,
            efmCuPme2BConstellation,
```

```
          efmCuPme2BProfileRowStatus
        }
        STATUS      current
        DESCRIPTION
          "A collection of objects that constitute a configuration
          profile for configuration of 2BaseTL ports."
        ::= { efmCuGroups 7 }

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

     efmCuPme10PStatusGroup OBJECT-GROUP
        OBJECTS {
          efmCuPme10PElectricalLength,
          efmCuPme10PFECCorrectedBlocks,
          efmCuPme10PFECUncorrectedBlocks
        }
        STATUS   current
        DESCRIPTION
          "A collection of objects providing status information
          specific to 10PassTS PMEs."
        ::= { efmCuGroups 9 }

    -- Compliance Statements

    efmCuCompliance MODULE-COMPLIANCE
        STATUS      current
        DESCRIPTION
          "The compliance statement for 2BaseTL/10PassTS interfaces.
          Compliance with the following external compliance statements
          is prerequisite:

          MIB Module              Compliance Statement
          ----------              --------------------
          IF-MIB                  ifCompliance3
```

```
            IF-INVERTED-STACK-MIB  ifInvCompliance
            EtherLike-MIB          dot3Compliance2
            MAU-MIB                mauModIfCompl3"

       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 ieee2BaseTSR or ieee2BaseTLR 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 2BaseTL
          and 6dB for 10PassTS."

        -- EdNote: To be Continued

    ::= { efmCuCompliances 1 }
END
```

6. Security Considerations

There is a number of managed objects defined in this 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:

o Changing of efmCuPmeAdminSubType MAY lead to a potential locking
   of the link, as peer PMEs of the same sub-type cannot exchange
   handshake messages.

o Changing of efmCuPAFAdminState to enabled MAY lead to a potential
   locking of the link, if the peer Phy does not support PAF.

o Changing of efmCuPAFDiscoveryCode, before the discovery operation,
   MAY lead to a wrongful discovery, with possile multiple -O ports
   connecting to the same -R (both -O ports have the same Discovery
   register value) and similar cases.

o 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.

o 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.

o Removal of a PME from the operationally 'up' EFMCu port,
   aggregating several PMEs, MAY lead a link rate degradation

The user of this 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 this 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 evasdrop to an EFMCu transmission simply by
"listening" to a cross-talk from an EFMCu pair, especially if the
parameters of the EFMCu link in question are known. In such
environments it is important to control even 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 this MIB module.

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 this MIB module 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.

7. Acknowledgments

Not yet.

8. References

8.1 Normative References

[802.3ah] IEEE, "Draft amendment to - 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 - Media Access Control Parameters,
           Physical Layers and Management Parameters for subscriber
           access networks", IEEE Draft P802.3ah/D3.3, April 2004.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
           Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2570] Case, J., Mundy, R., Partain, D. and B. Stewart,
           "Introduction to Version 3 of the Internet-standard
           Network Management Framework", RFC 2570, April 1999.

[RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
           McCloghrie, K., Rose, M. and S. Waldbusser, "Structure of
           Management Information Version 2 (SMIv2)", STD 58, RFC
           2578, April 1999.

[RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
           McCloghrie, K., Rose, M. and S. Waldbusser, "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.

[RFC3410] Case, J., Mundy, R., Partain, D. and B. Stewart,
           "Introduction and Applicability Statements for
           Internet-Standard Management Framework", RFC 3410,
           December 2002.

8.2 Informative References

[I-D.ietf-adslmib-gshdslbis]
           Sikes, C., Ray, B. and R. Abbi, "Definitions of Managed
           Objects for G.SHDSL.BIS Lines",
           draft-ietf-adslmib-gshdslbis-02 (work in progress), June
           2004.

[I-D.ietf-adslmib-vdsl-ext-mcm]
           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", draft-ietf-adslmib-vdsl-ext-mcm-04 (work in
                    progress), May 2004.

     [I-D.ietf-hubmib-efm-epon-mib]
                    Khermosh, L., "Managed Objects for the Ethernet Passive
                    Optical Networks", draft-ietf-hubmib-efm-epon-mib-01 (work
                    in progress), April 2004.

     [I-D.ietf-hubmib-efm-mib]
                    Squire, M., "Ethernet in the First Mile (EFM) Common MIB",
                    draft-ietf-hubmib-efm-mib-01 (work in progress), June
                    2004.

     [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.

     [RFC3635]   Flick, J., "Definitions of Managed Objects for the
                    Ethernet-like Interface Types", RFC 3635, September 2003.

     [RFC3636]   Flick, J., "Definitions of Managed Objects for IEEE 802.3
                    Medium Attachment Units (MAUs)", RFC 3636, September 2003.

Author's Address

     Edward Beili
     Actelis Networks
     Bazel 25
     Petach-Tikva
     Israel

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

Intellectual Property Statement

   The IETF takes no position regarding the validity or scope of any
   Intellectual Property Rights or other rights that might be claimed to
   pertain to the implementation or use of the technology described in
   this document or the extent to which any license under such rights
   might or might not be available; nor does it represent that it has
   made any independent effort to identify any such rights. Information
   on the procedures with respect to rights in RFC documents can be
   found in BCP 78 and BCP 79.

   Copies of IPR disclosures made to the IETF Secretariat and any
   assurances of licenses to be made available, or the result of an
   attempt made to obtain a general license or permission for the use of
   such proprietary rights by implementers or users of this
   specification can be obtained from the IETF on-line IPR repository at
   http://www.ietf.org/ipr.

   The IETF invites any interested party to bring to its attention any
   copyrights, patents or patent applications, or other proprietary
   rights that may cover technology that may be required to implement
   this standard. Please address the information to the IETF at
   ietf-ipr@ietf.org.

Disclaimer of Validity

   This document and the information contained herein are provided on an
   "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS
   OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET
   ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED,
   INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE
   INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
   WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

   Copyright (C) The Internet Society (2004). This document is subject
   to the rights, licenses and restrictions contained in BCP 78, and
   except as set forth therein, the authors retain all their rights.

Acknowledgment

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