Pseudowire (PW) Management Information Base (MIB)
draft-ietf-pwe3-pw-mib-14

Status of this Memo

Copyright Notice

Abstract

   This memo defines an experimental portion of the Management
   Information Base for use with network management protocols in the
   Internet community. In particular, it describes managed objects for
   modeling of Pseudowire Edge-to-Edge services carried over a general
   Packet Switched Network.

Table of Contents

1. Introduction

   This memo defines a portion of the Management Information Base (MIB)
   for use with network management protocols in the Internet community.
   In particular, it defines a MIB module that can be used to manage
   pseudowire (PW) services for transmission over a packet Switched
   Network (PSN) [RFC3931] [RFC4447]. This MIB module provides generic
   management of PWs which is common to all types of PSN and PW services
   defined by the IETF PWE3 Working Group.

   Comments should be made directly to the PWE3 mailing list at
   pwe3@ietf.org.

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].

3. Conventions

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [BCP14].

   This document adopts the definitions, acronyms and mechanisms
   described in [RFC3985] and [RFC3916]. Unless otherwise stated, the
   mechanisms of [RFC3985] apply and will not be re-described here.

4. Co-Authors

   The individuals listed below are co-authors of this document.

      Dave Danenberg - Litchfield Communications

Sharon Mantin - Corrigent Systems

5. Overview

The PWE3 MIB Modules architecture provides a layered modular model
into which any supported emulated service can be connected to any
supported PSN type. This specific MIB module provides the glue for
mapping between the emulated service onto the native PSN service. As
such the defining of a PW emulated service requires the use of at
least three types of MIB modules.

Starting from the emulated service, the first type is service-
specific module, which is emulated signal type dependent. These
modules are defined in other documents.

The second type is this module, the PW-STD-MIB module, which
configures general parameters of the PW that are common to all types
of emulated services and PSN types.

The third type of modules is PSN-specific module. There is a
different module for each type of PSN. These modules associate the
PW with one or more "tunnels" that carry the service over the PSN.
These modules are defined in other documents.

6. Structure of the MIB Module

The MIB Module consists of five tables;

- The generic configuration and status monitoring objects which are
common to all service types and PSN types (pwTable).

- The PW Performance Current Table (pwPerfCurrentTable) contains PW
statistics for the current 15-minute period.

- The PW Performance Interval Table (pwPerfIntervalTable) contains PW
statistics for historical intervals (usually 96 15-minute entries to
cover a 24 hour period).

- The PW Performance one day Interval Table (pwPerf1DayIntervalTable)
contains PW statistics for historical intervals accumulated per day.
Usually 30 1-Day entries to cover a monthly period.

- The mapping table (pwIndexMappingTable) enables the reverse mapping
of unique PWid parameters [peer IP, PW type and PW ID] and the
pwIndex.

   - The mapping table (pwGenFecIndexMappingTable) enables the reverse
   mapping of unique PWid parameters used in genFecSignaling
   [pwGroupAttachmentID, pwLocalAttachmentID and pwPeerAttachmentID] and
   the pwIndex.

   This MIB module uses TCs from [RFC2578], [RFC2579], [RFC2580],
   [RFC2863], [RFC3411], [RFC3593], [RFC3705], [RFC4001] and [PWTC], and
   references [RFC3413], [RFC4623] and [RFC4720].

7. PW-STD-MIB Module Usage

   An entry in the PW table (pwTable) MUST exist for all PW types (ATM,
   FR, Ethernet, SONET, etc.). This table holds generic parameters
   related to the PW creation and monitoring.

   A conceptual row can be created in the pwTable in one of the
   following ways:

   1) The operator creates a row in the PwTable when configuring the
   node for a new service. This mode MUST be supported by the agent,
   and MUST be used when creating a non-signaled (manually assigned) PW.

   2) The agent MAY create a row in the PwTable if a signaling message
   has been received from a peer node for a combination of signaling
   identifications parameters already unknown to the local node. This
   mode is OPTIONAL.

   3) The agent MAY create a row in the PwTable automatically due to
   some auto discovery application, or based on configuration that is
   done through non-SNMP applications. This mode is OPTIONAL.

   - The agent than creates the rows in the (locally supported)
   performance tables and reverse mapping tables in PW-STD-MIB module.

8. Relations to Other PWE3 MIB Modules

   - Based on the PSN type defined for the PW, a row is created in PSN
   specific module (for example [PWMPLSMIB]) and associated to the PW
   table by the common pwIndex.

   - Based on the PW type defined for the PW, a row is created in
   service-specific module (for example [CEPMIB]) and associated to the
   PW table by the common pwIndex.

   - Unless all the necessary entries in the applicable tables have been
   created and all the parameters have been consistently configured in

those tables, signaling cannot be performed from the local node, and
the pwVcOperStatus should report 'notPresent'.

9. Relations to the IF-MIB

The PW in general is not an ifIndex [RFC2863] on its own, for agent
scalability reasons. The PW is typically associated via the PWE3 MIB
modules to an ifIndex the PW is emulating. This ifIndex may
represent a physical entity - for example a PW emulating a SONET path
as in CEP: The PW itself is not an ifIndex, however the PW-STD-CEP-
MIB module associates the PW to the ifIndex of the path to be
emulated. In some cases, the PW will be associated to an ifIndex
representing a virtual interface. An example is VPLS service where
the PW emulates a logical interface of a (logical) bridge. The
physical ports association to the VPLS service instance is defined in
non-PW MIB modules in this case.

Exception to the above MAY exist in some implementations, where it is
convenient to manage the PW as an ifIndex in the ifTable. A special
ifType to represent a PW virtual interface (exact number to be
assigned by IANA) will be used in the ifTable in this case.

When the PW is managed as an ifIndex, by default it SHOULD NOT be
stacked, i.e. this ifIndex SHOULD NOT be layered above the respective
PSN tunnel ifIndex or the attachment circuit ifIndex or the interface
carrying the attachment circuit.

Note that the ifIndex that carries the PW toward/from the PSN is in
general not explicitly configured via PWE3 MIB modules except in rare
cases. In most cases the PW is carried inside a PSN tunnel, and the
interfaces carrying the tunnel are specified in the related MIB
modules that control the PSN tunnels.

10. PW Notifications

This MIB module includes notifications for PW entering the up or down
state, in accordance with the guidelines for interface notifications
as described in [RFC2863]. Implementers should be aware that in many
systems it is desired to correlate notifications, such that
notifications will not be emitted if higher hierarchy (such as ports
or tunnels) notifications are already in effect. Specifically for
PWs, it is anticipated that most network's equipment failures turn
into lowerLayerDown state at the PW level, where higher hierarchy
level notification has already been emitted.

When a PW is represented as an ifIndex, it is RECOMMENDED that PW

notifications will be turned off, to avoid duplication with the
ifIndex status change notifications.

11. Example of the PW MIB Modules Usage

In this section we provide an example of using the MIB objects
described in section 7 to set up a CEP PW over MPLS PSN. While this
example is not meant to illustrate every permutation of the MIB, it
is intended as an aid to understanding some of the key concepts. It
is meant to be read after going through the MIB itself.

In this example a PW service for CEP is configured over an MPLS PSN
(MPLS-TE tunnel). It uses LDP as in [RFC4447] for service set-up.

For the operation in the service specific MIB modules and the PSN
specific MIB modules, see the specific MIB module memo. This example
is continued in the memo describing the PW-CEP-STD-MIB module (for
example [CEPMIB]) and the PW-MPLS-STD-MIB module [PWMPLSMIB]).

    In the PW-STD-MIB module:

    In pwTable:
    {
        pwIndex   5,

        pwType   cep,
        pwOwner   pwIdFecSignaling,
        pwPsnType   mpls,
        pwSetUpPriority   0, -- Highest
        pwHoldingPriority   0, -- Highest
        pwInboundMode   loose,

        pwPeerAddrType   ipv4,
        pwPeerAddr   192.0.2.5, -- In this case equal to the
                                         -- peer LDP entity IP addr
        pwID   10,
        pwLocalGroupID   12,
        ..

        pwCwPreference   true,   -- Actually ignored for CEP
        pwLocalIfMtu   0,   -- Do not send ifMtu parameter
        pwLocalIfString   false, -- Do not send interface string
        pwCapabAdvert   0,   -- Does not support status
                                       -- report to the peer.
        pwRemoteGroupID   0xFFFF, -- Will be received by
                                       -- signaling protocol
        pwRemoteCwStatus   notKnownYet,
        pwRemoteIfMtu   0,
        pwRemoteIfString   "",
        pwRemoteCapabilities   notYetKnown,
        ..
        pwOutboundVcLabel   0xFFFF, -- Will be received by
                                       -- signaling protocol
        pwInboundVcLabel   0xFFFF, -- Will be set by signaling
                                       -- protocol
        pwName   "Example of CEP PW",
        pwDescr   "",
        ..

        pwAdminStatus   up,
        ..
         }

12. IANA PWE3 MIB Module

This section contains the initial version of the IANA-PWE3-MIB. IANA
is requested to update this MIB module based on expert review as
defined in [RFC2434]. Each new assignment of PW type or PW PSN type
made by IANA based on the procedures described in [RFC4446] should be
documented in the online version of IANA-PWE3-MIB. The current IANA-
PWE3-MIB contains PW types as requested in [RFC4446] and [RFC4863].


```
    IANA-PWE3-MIB DEFINITIONS ::= BEGIN

    IMPORTS
        MODULE-IDENTITY, mib-2
           FROM SNMPv2-SMI -- [RFC2578]

        TEXTUAL-CONVENTION
           FROM SNMPv2-TC; -- [RFC2579]

    ianaPwe3MIB  MODULE-IDENTITY
        LAST-UPDATED "200712091200Z"  -- 9 December 2007 12:00:00 GMT
        ORGANIZATION "IANA"
        CONTACT-INFO
             "Internet Assigned Numbers Authority
              Internet Corporation for Assigned Names and Numbers
              4676 Admiralty Way, Suite 330
              Marina del Rey, CA 90292-6601

              Phone: +1 310 823 9358
              EMail: iana@iana.org"
        DESCRIPTION
             "This MIB module defines the IANAPwTypeTC and
              IANAPwPsnTypeTC textual conventions for use in PWE3
              MIB modules.

              Any additions or changes to the contents of this MIB
              module require either publication of an RFC, Designated
              Expert Review as defined in RFC 2434, Guidelines for
              Writing an IANA Considerations Section in RFCs, and should
              be based on the procedures defined in [RFC4446]. The
              Designated Expert will be selected by the IESG Area
              Director(s) of the internet Area.

              Copyright (C) The IETF Trust (2008).  The initial
              version of this MIB module was published in RFC yyyy;
              for full legal notices see the RFC itself.  Supplementary
              information may be available at:
```

        http://www.ietf.org/copyrights/ianamib.html."
     -- RFC Ed.: replace yyyy with actual RFC number & remove this note

      REVISION   "200712091200Z" -- 9 December 2007 12:00:00 GMT
      DESCRIPTION "Original version, published as part of RFC yyyy."
  -- RFC Editor: please fill the yyyy and remove this note.

      ::= { mib-2 XXXX }
  -- RFC Editor: please fill the XXXX based on the IANA allocation
  -- and remove this note.

   IANAPwTypeTC ::= TEXTUAL-CONVENTION
      STATUS   current
      DESCRIPTION
         "Indicates the PW type (i.e. the carried service). "
    SYNTAX   INTEGER {
       other(0),
       frameRelayDlciMartiniMode(1),
       atmAal5SduVcc(2),
       atmTransparent(3),
       ethernetTagged(4),
       ethernet(5),
       hdlc(6),
       ppp(7),
       cem(8), -- Historic type
       atmCellNto1Vcc(9),
       atmCellNto1Vpc(10),
       ipLayer2Transport(11),
       atmCell1to1Vcc(12),
       atmCell1to1Vpc(13),
       atmAal5PduVcc(14),
       frameRelayPortMode(15),
       cep(16),
       e1Satop(17),
       t1Satop(18),
       e3Satop(19),
       t3Satop(20),
       basicCesPsn(21),
       basicTdmIp(22),
       tdmCasCesPsn(23),
       tdmCasTdmIp(24),
       frDlci(25),
       wildcard (32767)
        }

   IANAPwPsnTypeTC ::= TEXTUAL-CONVENTION
      STATUS   current
      DESCRIPTION

```
        "Identifies the PSN type which the PW will use over the
         network."
      SYNTAX    INTEGER {
         mpls         (1),
         l2tp         (2),
         udpOverIp    (3),
         mplsOverIp   (4),
         mplsOverGre  (5),
         other        (6)
         }

  IANAPwCapabilities ::= TEXTUAL-CONVENTION
      STATUS       current
      DESCRIPTION
         "This TC describes a collection of capabilities related to
          a specific PW.
          Values may be added in the future based on new capabilities
          introduced in IETF documents.
          "
      SYNTAX    BITS {
        pwStatusIndication (0), -- Applicable only if maintenance
                                -- protocol is in use.
        pwVCCV              (1)
      }

   END
```

13. Object Definitions

```
  PW-STD-MIB DEFINITIONS ::= BEGIN

  IMPORTS
     NOTIFICATION-TYPE, MODULE-IDENTITY, OBJECT-TYPE,
     Integer32, Unsigned32, Counter32, Counter64, TimeTicks,
     transmission
        FROM SNMPv2-SMI                      -- [RFC2578]

     MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF                     -- [RFC2580]

     TruthValue, RowStatus, StorageType,
     TimeStamp
```

```
        FROM SNMPv2-TC                      -- [RFC2579]


    SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB         -- [RFC3411]


    InterfaceIndexOrZero
        FROM IF-MIB                     -- [RFC2863]


    InetAddressType, InetAddress
        FROM INET-ADDRESS-MIB           -- [RFC4001]


    PerfCurrentCount, PerfIntervalCount
        FROM PerfHist-TC-MIB            -- [RFC3593]


    HCPerfCurrentCount, HCPerfIntervalCount, HCPerfTimeElapsed,
    HCPerfValidIntervals
        FROM HC-PerfHist-TC-MIB         -- [RFC3705]


    PwIndexType, PwIndexOrZeroType, PwGroupID, PwIDType,
    PwOperStatusTC, PwAttachmentIdentifierType, PwCwStatusTC,
    PwStatus, PwFragSize, PwFragStatus, PwGenIdType
        FROM PW-TC-STD-MIB              -- [PWTC]
 -- RFC Editor: Please replace [PWTC] with RFC number and remove this
 -- note.


    IANAPwTypeTC, IANAPwPsnTypeTC, IANAPwCapabilities
        FROM IANA-PWE3-MIB                      -- Reference will be added
                                               -- When IANA will create the
                                               -- MIB module
 ;

 pwStdMIB MODULE-IDENTITY
    LAST-UPDATED "200711121200Z"  -- 12 November 2007 12:00:00 GMT
    ORGANIZATION "Pseudowire Edge-to-Edge Emulation (PWE3) Working
                  Group"
    CONTACT-INFO
        "
          David Zelig
          E-mail: davidz@corrigent.com

          Thomas D. Nadeau
          Email: thomas.nadeau@bt.com

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

    DESCRIPTION
```

          "This MIB module contains managed object definitions for
           pseudowire operation as in Bryant, S. and P. Pate, 'Pseudo
           Wire Emulation Edge-to-Edge (PWE3) Architecture' [RFC3985],
           Martini, L., et al, 'Pseudowire Setup and Maintenance Using
           the Label Distribution Protocol (LDP)' [RFC4447], and
           Townsley, M., et al, 'Layer Two Tunneling Protocol
           (Version 3)' [RFC3931].

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

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

           Copyright (C) The IETF Trust (2008). This version
           of this MIB module is part of RFC XXX;
           For full legal notices see the RFC itself or
           http://www.ietf.org/copyrights/ianamib.html
      -- RFC Ed.: replace XXX with actual RFC number & remove this note
           "

      -- Revision history.

      REVISION
          "200711121200Z" -- 12 November 2007 12:00:00 GMT
      DESCRIPTION "Initial version published as part of RFC YYYY."
    -- RFC Editor: please replace YYYY with IANA assigned value, and
    -- delete this note.

          ::= { transmission ZZZZ }
    -- RFC Editor: please replace ZZZZ with IANA assigned value, and
    -- delete this note.

    -- Top-level components of this MIB.

    -- Notifications
    pwNotifications OBJECT IDENTIFIER
                                  ::= { pwStdMIB 0 }
    -- Tables, Scalars
    pwObjects   OBJECT IDENTIFIER
                                  ::= { pwStdMIB 1 }
    -- Conformance
    pwConformance   OBJECT IDENTIFIER
                                  ::= { pwStdMIB 2 }

```
-- PW Virtual Connection Table

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

pwTable   OBJECT-TYPE
   SYNTAX        SEQUENCE OF PwEntry
   MAX-ACCESS    not-accessible
   STATUS        current
   DESCRIPTION
       "This table specifies information for configuring and
        status monitoring which are common to all service types
        and PSN types."
   ::= { pwObjects 2 }

pwEntry   OBJECT-TYPE
   SYNTAX        PwEntry
   MAX-ACCESS    not-accessible
   STATUS        current
   DESCRIPTION
       "A row in this table represents a pseudowire (PW) virtual
        connection across a packet network. It is indexed by
        pwIndex, which uniquely identifies a singular
        connection.
        A row can be created by an operator command from a
        management plan of a PE, by signaling or due to autodiscovery
        process. Operator's command can be issued via non SNMP
        application; in such case a row will be created implicitly
        by the agent.
        The read-create objects in this tables are divided into
        three categories:
        1) Objects that MUST NOT be changed after row activation.
        These are objects that define basic properties of the
        PW (for example type, destination, etc.).
        2) Objects that MAY be changed when the PW is
```

defined as not active. A change of these objects involves
re-signaling of the PW or it might be traffic affecting.
PW not active is defined as one of the following
conditions:
        a) The pwRowStatus is notInService(2).
        b) The pwRowStatus is notReady(3).
        c) The pwAdminStatus is down(2).
If the operator needs to change one of the values for an
active row, the operator can either set the pwRowStatus to
notInService(2) or set pwAdminStatus to down(2).
Signaling (or traffic) is initiated again upon setting
the pwRowStatus to active(1) or setting the pwAdminStatus
to up(1) or testing(3) respectively.
3) Objects that MAY be changed at any time.

A PW MAY have an entry in the ifTable in addition to the
entry in this table. In this case a special ifType for PW
will be set in the ifTable, and the ifIndex in the ifTable
of the PW will be set in the pwIfIndex object in this table.

By default, all the read-create objects MUST NOT be
changed after row activation, unless specifically indicated
in the individual object description.

Manual entries in this table SHOULD be preserved after a
reboot, the agent MUST ensure the integrity of those
entries. If the set of entries of a specific row are found to
be non consistent after reboot, the PW pwOperStatus MUST be
declared as notPresent(5).
        "
    INDEX { pwIndex }

          ::= { pwTable 1 }

  PwEntry ::= SEQUENCE {
        pwIndex   PwIndexType,
        pwType   IANAPwTypeTC,
        pwOwner   INTEGER,
        pwPsnType   IANAPwPsnTypeTC,
        pwSetUpPriority   Integer32,
        pwHoldingPriority   Integer32,
        pwPeerAddrType   InetAddressType,
        pwPeerAddr   InetAddress,
        pwAttachedPwIndex   PwIndexOrZeroType,
        pwIfIndex   InterfaceIndexOrZero,

        pwID   PwIDType,
        pwLocalGroupID   PwGroupID,

```
        pwGroupAttachmentID   PwAttachmentIdentifierType,
        pwLocalAttachmentID   PwAttachmentIdentifierType,
        pwRemoteAttachmentID    PwAttachmentIdentifierType,


        pwCwPreference        TruthValue,
        pwLocalIfMtu          Unsigned32,


        pwLocalIfString       TruthValue,
        pwLocalCapabAdvert    IANAPwCapabilities,
        pwRemoteGroupID       PwGroupID,
        pwCwStatus            PwCwStatusTC,
        pwRemoteIfMtu         Unsigned32,
        pwRemoteIfString      SnmpAdminString,
        pwRemoteCapabilities  IANAPwCapabilities,

        pwFragmentCfgSize     PwFragSize,
        pwRmtFragCapability   PwFragStatus,
        pwFcsRetentionCfg     INTEGER,
        pwFcsRetentionStatus  BITS,

        pwOutboundLabel       Unsigned32,
        pwInboundLabel        Unsigned32,

        pwName                SnmpAdminString,
        pwDescr               SnmpAdminString,
        pwCreateTime          TimeStamp,
        pwUpTime              TimeTicks,
        pwLastChange          TimeTicks,
        pwAdminStatus         INTEGER,
        pwOperStatus          PwOperStatusTC,
        pwLocalStatus         PwStatus,
        pwRemoteStatusCapable INTEGER,
        pwRemoteStatus        PwStatus,
        pwTimeElapsed         HCPerfTimeElapsed,
        pwValidIntervals      HCPerfValidIntervals,
        pwRowStatus           RowStatus,
        pwStorageType         StorageType,
        pwOamEnable           TruthValue,
        pwGenAGIType          PwGenIdType,
        pwGenLocalAIIType     PwGenIdType,
        pwGenRemoteAIIType    PwGenIdType
      }

  pwIndex OBJECT-TYPE
     SYNTAX        PwIndexType
     MAX-ACCESS    not-accessible
     STATUS        current
```

```
      DESCRIPTION
          "A unique index for the conceptual row identifying a PW within
           this table."
      ::= { pwEntry 1 }

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

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

  pwPsnType OBJECT-TYPE
     SYNTAX   IANAPwPsnTypeTC
     MAX-ACCESS    read-create
     STATUS        current
     DESCRIPTION
          "This object is set by the operator to indicate the PSN type.
           Based on this object, the relevant PSN table's entry is
```

             created in the PSN specific MIB modules.
             "
        ::= { pwEntry 4 }

   pwSetUpPriority OBJECT-TYPE
        SYNTAX   Integer32 (0..7)
        MAX-ACCESS   read-create
        STATUS   current
        DESCRIPTION
             "This object defines the relative priority of the PW
              during set-up in a lowest-to-highest fashion, where 0
              is the highest priority. PWs with the same priority
              are treated with equal priority. PW that have not yet
              succeeded to set-up will report 'dormant' in the
              pwOperStatus.
              This value is significant if there are competing resources
              among PWs and the implementation support this feature.
              Equal priority handling with competing resources is
              implementation specific.
              This object MAY be changed at any time."
        DEFVAL { 0 }
        ::= { pwEntry 5 }

   pwHoldingPriority OBJECT-TYPE
        SYNTAX   Integer32 (0..7)
        MAX-ACCESS   read-create
        STATUS   current
        DESCRIPTION
             "This object defines the relative holding priority of the
              PW in a lowest-to-highest fashion, where 0 is the highest
              priority. PWs with the same priority are treated equally.
              This value is significant if there are competing resources
              among PWs and the implementation support this feature.
              Equal priority handling with competing resources is
              implementation specific.
              This object MAY be changed only if the PW is not active."
        DEFVAL { 0 }
        ::= { pwEntry 6 }

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

       ::= { pwEntry 8 }

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

   pwAttachedPwIndex OBJECT-TYPE
      SYNTAX   PwIndexOrZeroType
      MAX-ACCESS   read-create
      STATUS   current
      DESCRIPTION
          "If the PW is attached to another PW instead of a local
           native service, this item indicates the pwIndex of the
           attached PW. Otherwise, this object MUST
           be set to zero. Attachment to another PW will have no
           PW specific entry in any of the service MIB modules. "
      DEFVAL { 0 }
      ::= { pwEntry 10 }

   pwIfIndex OBJECT-TYPE
      SYNTAX   InterfaceIndexOrZero
      MAX-ACCESS   read-create
      STATUS   current
      DESCRIPTION
          "This object indicates the ifIndex of the PW if the PW is
           represented in the ifTable. Otherwise, it MUST be set
           to zero."
      DEFVAL { 0 }
      ::= { pwEntry 11 }

   pwID OBJECT-TYPE
      SYNTAX   PwIDType
      MAX-ACCESS   read-create
      STATUS   current
      DESCRIPTION
          "Pseudowire identifier.

           If the pwOwner object is 'pwIdFecSignaling' or
           'l2tpControlProtocol', then this object is signaled in the
           outgoing PW ID field within the 'Virtual Circuit FEC
           Element'. For other values of pwOwner, this object is not

```
            signaled and it MAY be set to zero.

            For implementations that support the pwIndexMappingTable,
            a non-zero value is RECOMMENDED, even if this
            identifier is not signaled.  This is so that reverse
            mappings can be provided by pwIndexMappingTable and
            pwPeerMappingTable.  It is therefore RECOMMENDED that the
            value of this pwID be unique (or if pwPeerAddrType is not
            'unknown', at least [pwType,pwID,pwPeerAddrType,pwPeerAddr]
            is unique.)"
        REFERENCE
            "Martini, et al, 'Pseudowire Setup and Maintenance using
             the Label Distribution Protocol', RFC 4447."
       ::= { pwEntry 12 }

   pwLocalGroupID OBJECT-TYPE
      SYNTAX         PwGroupID
      MAX-ACCESS     read-create
      STATUS         current
      DESCRIPTION
          "Used in the Group ID field sent to the peer PWES
           within the maintenance protocol used for PW setup.
           It SHOULD be set to zero if maintenance protocol is
           not used."
      REFERENCE
          "Martini, et al, 'Pseudowire Setup and Maintenance using
           the Label Distribution Protocol', RFC 4447."
      ::= { pwEntry 13 }

   pwGroupAttachmentID OBJECT-TYPE
      SYNTAX         PwAttachmentIdentifierType
      MAX-ACCESS     read-create
      STATUS         current
      DESCRIPTION
          "This object is an octet string representing the attachment
           group identifier (AGI) that this PW belongs too, which
           typically identifies the VPN ID.
           Applicable if pwOwner equal 'genFecSignaling'."
      REFERENCE
          "Martini, et al, 'Pseudowire Setup and Maintenance using
           the Label Distribution Protocol', RFC 4447."
      ::= { pwEntry 14 }

   pwLocalAttachmentID   OBJECT-TYPE
      SYNTAX         PwAttachmentIdentifierType
      MAX-ACCESS     read-create
      STATUS         current
      DESCRIPTION
```

```
             "This object is an octet string representing the local
              forwarder attachment individual identifier (AII) to be
              used by this PW. It is used as the SAII for outgoing
              signaling messages and the TAII in the incoming messages
              from the peer.
              Applicable if pwOwner equal 'genFecSignaling'."
         REFERENCE
             "Martini, et al, 'Pseudowire Setup and Maintenance using
              the Label Distribution Protocol', RFC 4447."
         ::= { pwEntry 15 }

     pwRemoteAttachmentID   OBJECT-TYPE
        SYNTAX         PwAttachmentIdentifierType
        MAX-ACCESS     read-create
        STATUS         current
        DESCRIPTION
             "This object is an octet string representing the remote
              forwarder attachment individual identifier (AII) to be
              used by this PW. It is used as the TAII for outgoing
              signaling messages and the SAII in the incoming messages
              from the peer.
              Applicable if pwOwner equal 'genFecSignaling'."
         REFERENCE
             "Martini, et al, 'Pseudowire Setup and Maintenance using
              the Label Distribution Protocol', RFC 4447."
         ::= { pwEntry 16 }

     pwCwPreference OBJECT-TYPE
        SYNTAX         TruthValue
        MAX-ACCESS     read-create
        STATUS         current
        DESCRIPTION
             "Defines if the control word will be sent with each packet
              by the local node. Some PW types mandate the use of a
              control word, and in such cases the value configured for
              this object has no effect on the existence of the control
              word.
              This object MAY be changed only if the PW is not active."
         REFERENCE
             "Martini, et al, 'Pseudowire Setup and Maintenance using
              the Label Distribution Protocol.', RFC 4447."
         DEFVAL { false }
         ::= { pwEntry 17 }

     pwLocalIfMtu OBJECT-TYPE
        SYNTAX         Unsigned32 (0..65535)
        MAX-ACCESS     read-create
        STATUS         current
```

          DESCRIPTION
              "If not equal to zero, the optional IfMtu object in the
               signaling protocol will be sent with this value, which
               represents the locally-supported MTU size over the
               interface (or the virtual interface) associated with the
               PW.
               This object MAY be changed only if the PW is not active."
          REFERENCE
              "Martini, et al, 'Pseudowire Setup and Maintenance using
               the Label Distribution Protocol', RFC 4447."
          DEFVAL { 0 }
          ::= { pwEntry 18 }

     pwLocalIfString OBJECT-TYPE
          SYNTAX   TruthValue
          MAX-ACCESS   read-create
          STATUS   current
          DESCRIPTION
              "A PW MAY be associated to an interface (or a virtual
               interface) in the ifTable of the node as part of the
               service configuration. This object defines if the
               maintenance protocol will send the interface's name
               (ifAlias) as appears in the ifTable. If set to false,
               the optional element will not be sent.
               This object MAY be changed only if the PW is not active."
          REFERENCE
              "Martini, et al, 'Pseudowire Setup and Maintenance using
               the Label Distribution Protocol', RFC 4447, section 5.5."
          DEFVAL { false }
          ::= { pwEntry 19 }

     pwLocalCapabAdvert OBJECT-TYPE
          SYNTAX   IANAPwCapabilities
          MAX-ACCESS   read-create
          STATUS   current
          DESCRIPTION
              "If maintenance protocol is used, it indicates the
               capabilities the local node will advertise to the
               peer. The operator MAY selectively assign partial set
               of capabilities. In case of manual configuration of the PW,
               the operator SHOULD set non conflicting options (for example
               only a single type of OAM) out of the available options
               in the implementation.
               It is possible to change the value of this object when the
               PW is not active. The agent MUST reject any attempt to set
               a capability that is not supported.

               The default value MUST be the full set of local node

```
        capabilities."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol', RFC 4447."
    ::= { pwEntry 20 }

  pwRemoteGroupID OBJECT-TYPE
    SYNTAX          PwGroupID
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This object is obtained from the Group ID field as
         received via the maintenance protocol used for PW setup.
         Value zero will be reported if not used.
         Value of 0xFFFFFFFF shall be used if the object is yet to be
         defined by the PW maintenance protocol."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol', RFC 4447."
    ::= { pwEntry 21 }

  pwCwStatus OBJECT-TYPE
    SYNTAX          PwCwStatusTC
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "If signaling is used for PW establishment, this object
         indicates the status of the control word negotiation,
         and in both; signaling or manual configuration it indicates
         if CW is to be present for this PW."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol', RFC 4447."
    ::= { pwEntry 22 }

  pwRemoteIfMtu OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "The remote interface MTU as (optionally) received from the
         remote node via the maintenance protocol. The object SHOULD
         report zero if MTU is not available."
    REFERENCE
        "Martini, et al, 'Pseudowire Setup and Maintenance using
         the Label Distribution Protocol', RFC 4447."
    ::= { pwEntry 23 }
```

```
pwRemoteIfString OBJECT-TYPE
   SYNTAX        SnmpAdminString (SIZE (0..80))
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
       "Indicates the interface description string as received by
        the maintenance protocol. It MUST be a NULL string if
        maintenance protocol is not used or the value is not known
        yet."
   REFERENCE
       "Martini, et al, 'Pseudowire Setup and Maintenance using
        the Label Distribution Protocol', RFC 4447, section 5.5."
   ::= { pwEntry 24 }

pwRemoteCapabilities OBJECT-TYPE
   SYNTAX        IANAPwCapabilities
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
       "Indicates the capabilities as received from the peer."
   REFERENCE
       "Martini, et al, 'Pseudowire Setup and Maintenance using
        the Label Distribution Protocol', RFC 4447."
   ::= { pwEntry 25 }

pwFragmentCfgSize OBJECT-TYPE
   SYNTAX        PwFragSize
   UNITS         "bytes"
   MAX-ACCESS    read-create
   STATUS        current
   DESCRIPTION
       "If set to a value other than zero, indicates that
        fragmentation is desired for this PW.
        This object MAY be changed only if the PW is not active."
   REFERENCE
       "Malis A., Townsley M., 'PWE3 Fragmentation and Reassembly',
        RFC 4623."
   DEFVAL { 0 } -- i.e. fragmentation not desired
   ::= { pwEntry 26 }

pwRmtFragCapability OBJECT-TYPE
   SYNTAX        PwFragStatus
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
       "The status of the fragmentation based on the local
        configuration and the peer capabilities as received from
        the peer when control protocol is used."
```

```
      REFERENCE
          "Malis A., Townsley M., 'PWE3 Fragmentation and Reassembly',
           RFC 4623."
      ::= { pwEntry 27 }

   pwFcsRetentionCfg OBJECT-TYPE
      SYNTAX        INTEGER {
                    fcsRetentionDisable  (1),
                    fcsRetentionEnable   (2)
      }
      MAX-ACCESS    read-create
      STATUS        current
      DESCRIPTION
          "The local configuration of FCS retention for this PW. FCS
           retention can be configured for PW types HDLC, PPP and
           Ethernet only. If the implementation does not support
           FCS retention, error MUST be reported in pwFcsRetentionStatus.
           This object MAY be changed only if the PW is not active."
      REFERENCE
          "Malis A., et al.,  'PWE3 Frame Check Sequence Retention',
           RFC 4720."
      DEFVAL { fcsRetentionDisable }
      ::= { pwEntry 28 }

   pwFcsRetentionStatus OBJECT-TYPE
      SYNTAX    BITS {
                remoteIndicationUnknown     (0),
                remoteRequestFcsRetention   (1),
                fcsRetentionEnabled         (2),
                fcsRetentionDisabled        (3),
                localFcsRetentionCfgErr     (4),
                fcsRetentionFcsSizeMismatch (5)
                }
      MAX-ACCESS    read-only
      STATUS        current
      DESCRIPTION
         "The status of the FCS retention negotiation process based on
          local configuration and the remote advertisement.

          remoteIndicationUnknown - set if a FEC has not been received
          from the remote.

          remoteRequestFcsRetention - indicates that the peer has
          requested for FCS retention. FCS retention will be used if
          the local node is capable and configured to use it for this
          PW.

          fcsRetentionEnabled - FCS retention is enabled (both peers
```

were configured for FCS retention for signaled PW, or the
local node is configured and capable for FCS retention for
manually assigned PW).

fcsRetentionDisabled - FCS retention is disabled (not
configured locally or not advertised by the peer).

localFcsRetentionCfgErr - Set if the local node has been
configured for FCS retention but is not capable to support
it.

fcsRetentionFcsSizeMismatch - Set if there is an FCS size
mismatch between the local and the peer node.
"
    REFERENCE
        "Malis A., et al., 'PWE3 Frame Check Sequence Retention',
         RFC 4720"
    ::= { pwEntry 29 }

  pwOutboundLabel OBJECT-TYPE
    SYNTAX   Unsigned32
    MAX-ACCESS   read-create
    STATUS   current
    DESCRIPTION
        "The PW label used in the outbound direction (i.e. toward
         the PSN). It might be set manually if pwOwner is 'manual',
         otherwise setting done automatically.
         For MPLS, MPLS over IP or MPLS over GRE PSN, it represents
         the 20 bits of PW tag, for L2TP it represents the 32 bits
         Session ID and for IP PSN it represents the destination
         UDP port number.
         If the label is not yet known (signaling in process), the
         object SHOULD return a value of 0xFFFFFFFF.
         For manual configuration, this object MAY be changed only
         if the PW is not active."
    ::= { pwEntry 30 }

  pwInboundLabel OBJECT-TYPE
    SYNTAX   Unsigned32
    MAX-ACCESS   read-create
    STATUS   current
    DESCRIPTION
        "The PW label used in the inbound direction (i.e. packets
         received from the PSN). It may be set manually if pwOwner
         is 'manual',otherwise setting done automatically.
         For MPLS, MPLS over IP or MPLS over GRE PSN, it represents
         the 20 bits of PW tag, for L2TP it represents the 32 bits
         Session ID and for IP PSN it represents the source

          UDP port number.
          If the label is not yet known (signaling in process), the
          object SHOULD return a value of 0xFFFFFFFF.
          For manual configuration, this object MAY be changed only
          if the PW is not active."
     ::= { pwEntry 31 }

  pwName OBJECT-TYPE
     SYNTAX   SnmpAdminString
     MAX-ACCESS   read-create
     STATUS   current
     DESCRIPTION
         "The canonical name assigned to the PW. This object MAY be
          changed at any time."
     ::= { pwEntry 32 }

  pwDescr OBJECT-TYPE
     SYNTAX   SnmpAdminString
     MAX-ACCESS   read-create
     STATUS   current
     DESCRIPTION
         "A textual string containing information about the PW.
          If there is no description this object contains a zero
          length string. This object MAY be changed at any time."
     ::= { pwEntry 33 }

  pwCreateTime OBJECT-TYPE
     SYNTAX   TimeStamp
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION
         "The value of sysUpTime at the time this PW was created."
     ::= { pwEntry 34 }

  pwUpTime OBJECT-TYPE
     SYNTAX   TimeTicks
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION
         "Specifies the time since last change of pwOperStatus to
          Up(1)."
     ::= { pwEntry 35 }

  pwLastChange OBJECT-TYPE
     SYNTAX   TimeTicks
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION

```
            "The value of sysUpTime at the time the PW entered
             its current operational state.  If the current state was
             entered prior to the last re-initialization of the local
             network management subsystem, then this object contains a
             zero value."
        ::= { pwEntry 36 }

   pwAdminStatus OBJECT-TYPE
      SYNTAX    INTEGER {
                    up(1),      -- ready to pass packets
                    down(2),
                    testing(3) -- in a test mode
      }
      MAX-ACCESS    read-create
      STATUS        current
      DESCRIPTION
          "The desired operational status of this PW. This object MAY
           be set at any time."
        ::= { pwEntry 37 }

   pwOperStatus OBJECT-TYPE
      SYNTAX        PwOperStatusTC
      MAX-ACCESS    read-only
      STATUS        current
      DESCRIPTION
          "This object indicates the operational status of the PW, it
           does not reflect the status of the CE bound interface.
           It is set to down only if pwNotForwarding,
           psnFacingPwRxFault, or psnFacingPwTxFault indications are
           set in pwLocalStatus or pwRemoteStatus.
           It indicates 'lowerLayerDown' if the only reason for
           not being in the 'up' state is either outer tunnel
           or physical layer down of the network side is in the down
           state.
           All other states are declared based on the description in
           the textual convention.
           "
        ::= { pwEntry 38 }

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

```
            indications through the control protocol.
            "
     ::= { pwEntry 39 }

  pwRemoteStatusCapable OBJECT-TYPE
     SYNTAX         INTEGER {
            notApplicable    (1),
            notYetKnown      (2),
            remoteCapable    (3),
            remoteNotCapable (4)
     }
     MAX-ACCESS     read-only
     STATUS         current
     DESCRIPTION
          "Indicates the remote node capability to advertise the
           PW status notification.
           notAppicable SHOULD be reported for manually set PW, or
           if the local node is not capable of accepting the status
           notification object.
           notYetKnown SHOULD be reported if the signaling protocol
           has not yet finished the process of capability
           determination.
           remoteCapable and remoteNotcapable SHOULD be reported
           based on the initial signaling exchange that has
           determined the remote node capability.
          "
     ::= { pwEntry 40 }

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

  pwTimeElapsed OBJECT-TYPE
      SYNTAX  HCPerfTimeElapsed
      MAX-ACCESS  read-only
      STATUS  current
      DESCRIPTION
           "The number of seconds, including partial seconds,
```

```
           that have elapsed since the beginning of the current
           interval measurement period."
     ::= { pwEntry 42 }

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

  pwRowStatus OBJECT-TYPE
     SYNTAX        RowStatus
     MAX-ACCESS    read-create
     STATUS        current
     DESCRIPTION
         "For creating, modifying, and deleting this row.
          This object MAY be changed at any time."
     ::= { pwEntry 44 }

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

  pwOamEnable  OBJECT-TYPE
     SYNTAX        TruthValue
     MAX-ACCESS    read-create
     STATUS        current
     DESCRIPTION
         "This variable indicates if OAM is enabled for this
          PW. It MAY be changed at any time."
     DEFVAL { true }
     ::= { pwEntry 46 }

  pwGenAGIType OBJECT-TYPE
     SYNTAX        PwGenIdType
     MAX-ACCESS    read-create
     STATUS        current
     DESCRIPTION
         "This variable indicates the AGI type if generalized FEC
```

```
         (129) is used for PW signaling or configuration. It SHOULD
         return the value of zero otherwise."
   DEFVAL { 0 }
   ::= { pwEntry 47 }


pwGenLocalAIIType OBJECT-TYPE
   SYNTAX        PwGenIdType
   MAX-ACCESS    read-create
   STATUS        current
   DESCRIPTION
       "This object is the type of the local forwarder
        attachment individual identifier (AII) to be used
        by this PW if generalized FEC (129) is used for PW
        signaling or configuration."
   DEFVAL { 0 }
   ::= { pwEntry 48 }


pwGenRemoteAIIType OBJECT-TYPE
   SYNTAX        PwGenIdType
   MAX-ACCESS    read-create
   STATUS        current
   DESCRIPTION
       "This object is the type of the remote forwarder
        attachment individual identifier (AII) to be used
        by this PW if generalized FEC (129) is used for PW
        signaling or configuration."
   DEFVAL { 0 }
   ::= { pwEntry 49 }



-- End of PW Virtual Connection Table

-- PW Performance Table.

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

pwPerfCurrentEntry OBJECT-TYPE
   SYNTAX        PwPerfCurrentEntry
   MAX-ACCESS    not-accessible
   STATUS        current
```

```
      DESCRIPTION
           "An entry in this table is created by the agent for
            every PW."
      INDEX  { pwIndex }
      ::= { pwPerfCurrentTable 1 }

   PwPerfCurrentEntry ::= SEQUENCE {
        pwPerfCurrentInHCPackets         HCPerfCurrentCount,
        pwPerfCurrentInHCBytes           HCPerfCurrentCount,
        pwPerfCurrentOutHCPackets        HCPerfCurrentCount,
        pwPerfCurrentOutHCBytes          HCPerfCurrentCount,
        pwPerfCurrentInPackets           PerfCurrentCount,
        pwPerfCurrentInBytes             PerfCurrentCount,
        pwPerfCurrentOutPackets          PerfCurrentCount,
        pwPerfCurrentOutBytes            PerfCurrentCount
        }

   pwPerfCurrentInHCPackets OBJECT-TYPE
      SYNTAX        HCPerfCurrentCount
      MAX-ACCESS    read-only
      STATUS        current
      DESCRIPTION
           "High capacity counter for number of packets received by
            the PW (from the PSN) in the current 15-minute interval.
            This is the 64 bit version of pwPerfCurrentInPackets,
            if pwPerfCurrentInHCPackets is supported according to
            the rules spelled out in RFC2863."
      ::= { pwPerfCurrentEntry 1 }

   pwPerfCurrentInHCBytes OBJECT-TYPE
      SYNTAX        HCPerfCurrentCount
      MAX-ACCESS    read-only
      STATUS        current
      DESCRIPTION
           "High capacity counter for number of bytes received by the
            PW (from the PSN) in the current 15-minute interval.
            This is the 64 bit version of pwPerfCurrentInBytes, if
            pwPerfCurrentInHCBytes is supported according to the
            rules spelled out in RFC2863."
      ::= { pwPerfCurrentEntry 2 }

   pwPerfCurrentOutHCPackets OBJECT-TYPE
      SYNTAX        HCPerfCurrentCount
      MAX-ACCESS    read-only
      STATUS        current
      DESCRIPTION
           "High capacity counter for number of packets forwarded by
            the PW (to the PSN) in the current 15-minute interval.
```

```
              This is the 64 bit version of pwPerfCurrentOutPackets,
              if pwPerfCurrentOutHCPackets is supported according to
              the rules spelled out in RFC2863."
       ::= { pwPerfCurrentEntry 3 }

  pwPerfCurrentOutHCBytes OBJECT-TYPE
     SYNTAX        HCPerfCurrentCount
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
           "High capacity counter for number of bytes forwarded by
            the PW (to the PSN) in the current 15-minute interval.
            This is the 64 bit version of pwPerfCurrentOutBytes,
            if pwPerfCurrentOutHCBytes is supported according to
            the rules spelled out in RFC2863."
       ::= { pwPerfCurrentEntry 4 }

  pwPerfCurrentInPackets OBJECT-TYPE
     SYNTAX        PerfCurrentCount
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
           "The counter for number of packets received by the PW (from
            the PSN) in the current 15-minute interval.
            This is the 32 bit version of pwPerfCurrentInHCPackets,
            if pwPerfCurrentInHCPackets is supported according to
            the rules spelled out in RFC2863."
       ::= { pwPerfCurrentEntry 5 }

  pwPerfCurrentInBytes OBJECT-TYPE
     SYNTAX        PerfCurrentCount
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
           "The counter for number of bytes received by the
            PW (from the PSN) in the current 15-minute interval.
            It MUST be equal to the least significant 32 bits of
            pwPerfCurrentInHCBytes, if pwPerfCurrentInHCBytes is
            supported according to the rules spelled out in RFC2863."
       ::= { pwPerfCurrentEntry 6 }

  pwPerfCurrentOutPackets OBJECT-TYPE
     SYNTAX        PerfCurrentCount
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
           "The counter for number of packets forwarded by
            the PW (to the PSN) in the current 15-minute interval.
```

```
        It MUST be equal to the least significant 32 bits of
        pwPerfCurrentOutHCPackets, if
        pwPerfCurrentOutHCPackets is supported according to the
        rules spelled out in RFC2863."
 ::= { pwPerfCurrentEntry 7 }

pwPerfCurrentOutBytes OBJECT-TYPE
   SYNTAX        PerfCurrentCount
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
        "The counter for number of bytes forwarded by
        the PW (to the PSN) in the current 15-minute interval.
        It MUST be equal to the least significant 32 bits of
        pwPerfCurrentOutHCBytes, if pwPerfCurrentOutHCBytes is
        supported according to the rules spelled out in RFC2863."
 ::= { pwPerfCurrentEntry 8 }

-- End of PW Perf current Table

-- PW Performance Interval Table.

pwPerfIntervalTable  OBJECT-TYPE
   SYNTAX        SEQUENCE OF PwPerfIntervalEntry
   MAX-ACCESS    not-accessible
   STATUS        current
   DESCRIPTION
        "This table provides per-PW performance information for
        each interval."
 ::= { pwObjects 4 }

pwPerfIntervalEntry OBJECT-TYPE
   SYNTAX        PwPerfIntervalEntry
   MAX-ACCESS    not-accessible
   STATUS        current
   DESCRIPTION
        "An entry in this table is created by the agent for every
        PW."
   INDEX  { pwIndex, pwPerfIntervalNumber }
 ::= { pwPerfIntervalTable 1 }

PwPerfIntervalEntry ::= SEQUENCE {
     pwPerfIntervalNumber              Integer32,
     pwPerfIntervalValidData           TruthValue,
     pwPerfIntervalTimeElapsed         HCPerfTimeElapsed,
     pwPerfIntervalInHCPackets         HCPerfIntervalCount,
     pwPerfIntervalInHCBytes           HCPerfIntervalCount,
     pwPerfIntervalOutHCPackets        HCPerfIntervalCount,
```

```
        pwPerfIntervalOutHCBytes          HCPerfIntervalCount,
        pwPerfIntervalInPackets           PerfIntervalCount,
        pwPerfIntervalInBytes             PerfIntervalCount,
        pwPerfIntervalOutPackets          PerfIntervalCount,
        pwPerfIntervalOutBytes            PerfIntervalCount
                        }

  pwPerfIntervalNumber OBJECT-TYPE
     SYNTAX  Integer32 (1..96)
     MAX-ACCESS  not-accessible
     STATUS  current
     DESCRIPTION
          "A number N, between 1 and 96, which identifies the
           interval for which the set of statistics is available.
           The interval identified by 1 is the most recently
           completed 15 minute interval, and the interval identified
           by N is the interval immediately preceding the one
           identified by N-1.
           The minimum range of N is 1 through 4. The default range
           is 1 to 32. The maximum range of N is 1 through 96. "
     REFERENCE
         "Tesink, K. 'Definitions of Managed Objects for the
          SONET/SDH Interface Type', RFC 2558"
     ::= { pwPerfIntervalEntry 1 }

  pwPerfIntervalValidData OBJECT-TYPE
     SYNTAX        TruthValue
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
        "This variable indicates if the data for this interval
         is valid."
     ::= { pwPerfIntervalEntry 2 }

  pwPerfIntervalTimeElapsed OBJECT-TYPE
     SYNTAX        HCPerfTimeElapsed
     MAX-ACCESS  read-only
     STATUS        current
     DESCRIPTION
        "The duration of this interval in seconds."
     ::= { pwPerfIntervalEntry 3 }

  pwPerfIntervalInHCPackets OBJECT-TYPE
     SYNTAX        HCPerfIntervalCount
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
          "High capacity counter for number of packets received by
```

          the PW (from the PSN) during the interval. This is the 64
          bit version of pwPerfIntervalInPackets, if
          pwPerfIntervalInHCPackets is supported according to the
          rules spelled out in RFC2863."
     ::= { pwPerfIntervalEntry 4 }

  pwPerfIntervalInHCBytes OBJECT-TYPE
     SYNTAX   HCPerfIntervalCount
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION
          "High capacity counter for number of bytes received by the
          PW (from the PSN) during the interval.
          This is the 64 bit version of pwPerfIntervalInBytes, if
          pwPerfIntervalInHCBytes is supported according to the
          rules spelled out in RFC2863."
     ::= { pwPerfIntervalEntry 5 }

  pwPerfIntervalOutHCPackets OBJECT-TYPE
     SYNTAX   HCPerfIntervalCount
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION
          "High capacity counter for number of packets forwarded by
          the PW (to the PSN) during the interval.
          This is the 64 bit version of pwPerfIntervalOutPackets,
          if pwPerfIntervalOutHCPackets is supported according to
          the rules spelled out in RFC2863."
     ::= { pwPerfIntervalEntry 6 }

  pwPerfIntervalOutHCBytes OBJECT-TYPE
     SYNTAX   HCPerfIntervalCount
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION
          "High capacity counter for number of bytes forwarded by
          the PW (to the PSN) during the interval.
          This is the 64 bit version of pwPerfIntervalOutBytes,
          if pwPerfIntervalOutHCBytes is supported according to
          the rules spelled out in RFC2863."
     ::= { pwPerfIntervalEntry 7 }

  pwPerfIntervalInPackets OBJECT-TYPE
     SYNTAX   PerfIntervalCount
     MAX-ACCESS   read-only
     STATUS   current
     DESCRIPTION
         "This value represents the number of packets received

          by this PW during the interval.
          It MUST be equal to the least significant 32 bits of
          pwPerfIntervalInHCPackets if pwPerfIntervalInHCPackets
          is supported according to the rules spelled out in
          RFC2863."
      ::= { pwPerfIntervalEntry 8 }

   pwPerfIntervalInBytes OBJECT-TYPE
      SYNTAX   PerfIntervalCount
      MAX-ACCESS   read-only
      STATUS   current
      DESCRIPTION
          "This value represents the number of bytes received
           by this PW during the interval.
           It MUST be equal to the least significant 32 bits of
           if pwPerfIntervalInHCBytes is supported according to
           the rules spelled out in RFC2863."
      ::= { pwPerfIntervalEntry 9 }

   pwPerfIntervalOutPackets OBJECT-TYPE
      SYNTAX   PerfIntervalCount
      MAX-ACCESS   read-only
      STATUS   current
      DESCRIPTION
          "This value represents the number of packets sent by this
           PW during the interval.
           It MUST be equal to the least significant 32 bits of
           pwPerfIntervalOutHCPackets if
           pwPerfIntervalOutHCPackets is supported according to the
           rules spelled out in RFC2863."
      ::= { pwPerfIntervalEntry 10 }

   pwPerfIntervalOutBytes OBJECT-TYPE
      SYNTAX   PerfIntervalCount
      MAX-ACCESS   read-only
      STATUS   current
      DESCRIPTION
          "This value represents the number of bytes sent by this
           PW during the interval.
           It MUST be equal to the least significant 32
           bits of pwPerfIntervalOutHCBytes
           if pwPerfIntervalOutHCBytes is supported according to
           the rules spelled out in RFC2863."
      ::= { pwPerfIntervalEntry 11 }

   -- End of PW Performance Interval Table

   -- PW Performance 1 Day Interval Table.

   pwPerf1DayIntervalTable OBJECT-TYPE
      SYNTAX   SEQUENCE OF PwPerf1DayIntervalEntry
      MAX-ACCESS   not-accessible
      STATUS   current
      DESCRIPTION
           "This table provides per-PW Performance information for
            the current day measurement and the previous days interval."
      ::= { pwObjects 5 }

   pwPerf1DayIntervalEntry OBJECT-TYPE
      SYNTAX   PwPerf1DayIntervalEntry
      MAX-ACCESS   not-accessible
      STATUS   current
      DESCRIPTION
           "An entry in this table is created by the agent for every
            PW."
      INDEX { pwIndex, pwPerf1DayIntervalNumber }

      ::= { pwPerf1DayIntervalTable 1 }

   PwPerf1DayIntervalEntry ::= SEQUENCE {
        pwPerf1DayIntervalNumber   Unsigned32,
        pwPerf1DayIntervalValidData   TruthValue,
        pwPerf1DayIntervalTimeElapsed   HCPerfTimeElapsed,
        pwPerf1DayIntervalInHCPackets   Counter64,
        pwPerf1DayIntervalInHCBytes   Counter64,
        pwPerf1DayIntervalOutHCPackets   Counter64,
        pwPerf1DayIntervalOutHCBytes   Counter64
        }

   pwPerf1DayIntervalNumber OBJECT-TYPE
      SYNTAX   Unsigned32(1..31)
      MAX-ACCESS not-accessible
      STATUS   current
      DESCRIPTION
        "History Data Interval number. Interval 1 is the current day
         measurement period, interval 2 is the most recent previous
         day, and interval 30 is 31 days ago. Intervals 3..31 are
         optional."
      ::= { pwPerf1DayIntervalEntry 1 }

   pwPerf1DayIntervalValidData OBJECT-TYPE
      SYNTAX   TruthValue
      MAX-ACCESS   read-only
      STATUS   current
      DESCRIPTION

```
        "This variable indicates if the data for this interval
         is valid."
   ::= { pwPerf1DayIntervalEntry 2 }

pwPerf1DayIntervalTimeElapsed OBJECT-TYPE
   SYNTAX        HCPerfTimeElapsed
   UNITS         "seconds"
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
     "The number of seconds in the 1-day interval over which the
      performance monitoring information is actually counted.
      This value will be the same as the interval duration except
      in a situation where performance monitoring data could not
      be collected for any reason or agent clock adjustments."
   ::= { pwPerf1DayIntervalEntry 3 }

pwPerf1DayIntervalInHCPackets OBJECT-TYPE
   SYNTAX        Counter64
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
        "High capacity counter for the total number of packets
         received by the PW (from the PSN)."
   ::= { pwPerf1DayIntervalEntry 4 }

pwPerf1DayIntervalInHCBytes OBJECT-TYPE
   SYNTAX        Counter64
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
        "High capacity counter for the total number of bytes
         received by the PW (from the PSN)."
   ::= { pwPerf1DayIntervalEntry 5 }

pwPerf1DayIntervalOutHCPackets OBJECT-TYPE
   SYNTAX        Counter64
   MAX-ACCESS    read-only
   STATUS        current
   DESCRIPTION
        "High capacity counter for the total number of packets
         forwarded by the PW (to the PSN)."
   ::= { pwPerf1DayIntervalEntry 6 }

pwPerf1DayIntervalOutHCBytes OBJECT-TYPE
   SYNTAX        Counter64
   MAX-ACCESS    read-only
   STATUS        current
```

```
      DESCRIPTION
           "High capacity counter for the total number of bytes
            forwarded by the PW (to the PSN)."
      ::= { pwPerf1DayIntervalEntry 7 }

  -- End of PW Perf 1 Day Interval Table

  -- Error counter scalar

  pwPerfTotalErrorPackets OBJECT-TYPE
     SYNTAX         Counter32
     MAX-ACCESS     read-only
     STATUS         current
     DESCRIPTION
           "Counter for number of error at the PW processing level,
            for example packets received with unknown PW label."
      ::= { pwObjects 6 }

  -- Reverse mapping tables

  -- The PW ID mapping table
  pwIndexMappingTable  OBJECT-TYPE
     SYNTAX         SEQUENCE OF PwIndexMappingEntry
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
           "This table enables the reverse mapping the unique PWid
            parameters [peer IP, PW type and PW ID] and the
            pwIndex. The table is not applicable for PW created
            manually or by using the generalized FEC."
      ::= { pwObjects 7 }

  pwIndexMappingEntry OBJECT-TYPE
     SYNTAX         PwIndexMappingEntry
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
           "An entry in this table MUST be created by the agent for
            every PW created by the pwTable for which pwOwner
            equals pwIdFecSignaling and pwID is not zero.

            Implementers need to be aware that if the value of
            the pwIndexMappingPeerAddr (an OID) has more than
            113 sub-identifiers, then OIDs of column  instances
            in this table will have more than 128 sub-identifiers
            and cannot be accessed using SNMPv1, SNMPv2c, or SNMPv3."
      INDEX  { pwIndexMappingPwType,  pwIndexMappingPwID,
              pwIndexMappingPeerAddrType, pwIndexMappingPeerAddr
```

```
                    }
        ::= { pwIndexMappingTable 1 }

   PwIndexMappingEntry ::= SEQUENCE {
          pwIndexMappingPwType        IANAPwTypeTC,
          pwIndexMappingPwID          PwIDType,
          pwIndexMappingPeerAddrType  InetAddressType,
          pwIndexMappingPeerAddr      InetAddress,
          pwIndexMappingPwIndex       PwIndexType
                          }

   pwIndexMappingPwType OBJECT-TYPE
      SYNTAX         IANAPwTypeTC
      MAX-ACCESS     not-accessible
      STATUS         current
      DESCRIPTION
           "The PW type (indicates the service) of this PW."
      ::= { pwIndexMappingEntry 1 }

   pwIndexMappingPwID OBJECT-TYPE
      SYNTAX         PwIDType
      MAX-ACCESS     not-accessible
      STATUS         current
      DESCRIPTION
           "The PW ID of this PW. Zero if the PW is configured
            manually."
      ::= { pwIndexMappingEntry 2 }

   pwIndexMappingPeerAddrType OBJECT-TYPE
      SYNTAX         InetAddressType
      MAX-ACCESS     not-accessible
      STATUS         current
      DESCRIPTION
           "IP address type of the peer node."
      ::= { pwIndexMappingEntry 3 }

   pwIndexMappingPeerAddr OBJECT-TYPE
      SYNTAX         InetAddress
      MAX-ACCESS     not-accessible
      STATUS         current
      DESCRIPTION
           "IP address of the peer node."
      ::= { pwIndexMappingEntry 4 }

   pwIndexMappingPwIndex  OBJECT-TYPE
      SYNTAX         PwIndexType
      MAX-ACCESS     read-only
      STATUS         current
```

      DESCRIPTION
           "The value that represents the PW in the pwTable."
      ::= { pwIndexMappingEntry 5 }

  -- End of the PW ID mapping table

  -- The peer mapping table

  pwPeerMappingTable OBJECT-TYPE
     SYNTAX   SEQUENCE OF PwPeerMappingEntry
     MAX-ACCESS   not-accessible
     STATUS   current
     DESCRIPTION
           "This table provides reverse mapping of the existing PW
            based on PW type and PW ID ordering. This table is
            typically useful for EMS ordered query of existing PWs."
      ::= { pwObjects 8 }

  pwPeerMappingEntry OBJECT-TYPE
     SYNTAX   PwPeerMappingEntry
     MAX-ACCESS   not-accessible
     STATUS   current
     DESCRIPTION
           "An entry in this table is created by the agent for every
            PW entry in pwTable.

           Implementers need to be aware that if the value of the
           pwPeerMappingPeerAddr (an OID) has more than 113
           sub-identifiers, then OIDs of column instances in this
           table will have more than 128 sub-identifiers and cannot
           be accessed using SNMPv1, SNMPv2c, or SNMPv3."
     INDEX { pwPeerMappingPeerAddrType, pwPeerMappingPeerAddr,
              pwPeerMappingPwType, pwPeerMappingPwID }

      ::= { pwPeerMappingTable 1 }

  PwPeerMappingEntry ::= SEQUENCE {
        pwPeerMappingPeerAddrType   InetAddressType,
        pwPeerMappingPeerAddr   InetAddress,
        pwPeerMappingPwType   IANAPwTypeTC,
        pwPeerMappingPwID   PwIDType,
        pwPeerMappingPwIndex   PwIndexType
                       }

  pwPeerMappingPeerAddrType OBJECT-TYPE
     SYNTAX   InetAddressType
     MAX-ACCESS   not-accessible
     STATUS   current

```
     DESCRIPTION
         "IP address type of the peer node."
     ::= { pwPeerMappingEntry 1 }

  pwPeerMappingPeerAddr OBJECT-TYPE
     SYNTAX        InetAddress
     MAX-ACCESS    not-accessible
     STATUS        current
     DESCRIPTION
         "IP address of the peer node."
     ::= { pwPeerMappingEntry 2 }

  pwPeerMappingPwType OBJECT-TYPE
     SYNTAX        IANAPwTypeTC
     MAX-ACCESS    not-accessible
     STATUS        current
     DESCRIPTION
         "The PW type (indicates the emulated service) of this PW."
     ::= { pwPeerMappingEntry 3 }

  pwPeerMappingPwID OBJECT-TYPE
     SYNTAX        PwIDType
     MAX-ACCESS    not-accessible
     STATUS        current
     DESCRIPTION
         "The PW ID of this PW. Zero if the PW is configured
          manually."
     ::= { pwPeerMappingEntry 4 }

  pwPeerMappingPwIndex  OBJECT-TYPE
     SYNTAX        PwIndexType
     MAX-ACCESS    read-only
     STATUS        current
     DESCRIPTION
        "The value that represents the PW in the pwTable."
     ::= { pwPeerMappingEntry 5 }

  -- End of the peer mapping table

  -- End of reverse mapping tables

  pwUpDownNotifEnable  OBJECT-TYPE
     SYNTAX      TruthValue
     MAX-ACCESS  read-write
     STATUS      current
     DESCRIPTION
        "If this object is set to true(1), then it enables
         the emission of pwUp and pwDown
```

          notifications; otherwise these notifications are not
          emitted."
      REFERENCE
         "See also [RFC3413] for explanation that
          notifications are under the ultimate control of the
          MIB module in this document."
      DEFVAL { false }
      ::= { pwObjects 9 }

  pwDeletedNotifEnable OBJECT-TYPE
      SYNTAX   TruthValue
      MAX-ACCESS   read-write
      STATUS   current
      DESCRIPTION
         "If this object is set to true(1), then it enables the
          emission of pwDeleted notification; otherwise this
          notification is not emitted."
      REFERENCE
         "See also [RFC3413] for explanation that
          notifications are under the ultimate control of the
          MIB module in this document."
      DEFVAL { false }
      ::= { pwObjects 10 }

  pwNotifRate OBJECT-TYPE
      SYNTAX   Unsigned32
      MAX-ACCESS   read-write
      STATUS   current
      DESCRIPTION
         "This object defines the maximum number of PW notifications
          that can be emitted from the device per second."
      ::= { pwObjects 11 }

  -- The Gen Fec PW ID mapping table

  pwGenFecIndexMappingTable OBJECT-TYPE
      SYNTAX   SEQUENCE OF PwGenFecIndexMappingEntry
      MAX-ACCESS   not-accessible
      STATUS   current
      DESCRIPTION
          "This table enables the reverse mapping the unique PWid
           parameters [GroupAttachmentID, LocalAttachmentID and
           PeerAttachmentID] and the pwIndex. The table is only
           applicable for PW using the generalized FEC."
      ::= { pwObjects 12 }

  pwGenFecIndexMappingEntry OBJECT-TYPE
      SYNTAX   PwGenFecIndexMappingEntry

```
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "An entry in this table MUST be created by the agent for
           every PW created by the pwTable for which pwOwner
           equals genFecSignaling.

           Implementors need to be aware that if the combined value
           of pwGenFecIndexMappingAGI,  pwGenFecIndexMappingLocalAII,
           and pwGenFecIndexMappingRemoteAII (OIDs) has more than
           113 sub-identifiers, then OIDs of column instances
           in this table will have more than 128  sub-identifiers
           and cannot be accessed using SNMPv1, SNMPv2c, or SNMPv3."
     INDEX  { pwGenFecIndexMappingAGIType,
              pwGenFecIndexMappingAGI,
              pwGenFecIndexMappingLocalAIIType,
              pwGenFecIndexMappingLocalAII,
              pwGenFecIndexMappingRemoteAIIType,
              pwGenFecIndexMappingRemoteAII
              }
     ::= { pwGenFecIndexMappingTable 1 }

  PwGenFecIndexMappingEntry ::= SEQUENCE {
     pwGenFecIndexMappingAGIType         PwGenIdType,
     pwGenFecIndexMappingAGI             PwAttachmentIdentifierType,
     pwGenFecIndexMappingLocalAIIType    PwGenIdType,
     pwGenFecIndexMappingLocalAII        PwAttachmentIdentifierType,
     pwGenFecIndexMappingRemoteAIIType   PwGenIdType,
     pwGenFecIndexMappingRemoteAII       PwAttachmentIdentifierType,
     pwGenFecIndexMappingPwIndex         PwIndexType
  }

  pwGenFecIndexMappingAGIType OBJECT-TYPE
     SYNTAX         PwGenIdType
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "This object is the type of the attachment
           group identifier (AGI) that this PW belongs too."
     ::= { pwGenFecIndexMappingEntry 1 }

  pwGenFecIndexMappingAGI OBJECT-TYPE
     SYNTAX         PwAttachmentIdentifierType
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "This object is an octet string representing the attachment
           group identifier (AGI) that this PW belongs too,
```

```
            which typically identifies the VPN ID."
     ::= { pwGenFecIndexMappingEntry 2 }


  pwGenFecIndexMappingLocalAIIType OBJECT-TYPE
     SYNTAX         PwGenIdType
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "this object is the type of the local forwarder
           attachment individual identifier (AII) to be used
           by this PW."
     ::= { pwGenFecIndexMappingEntry 3 }

  pwGenFecIndexMappingLocalAII OBJECT-TYPE
     SYNTAX         PwAttachmentIdentifierType
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "This object is an octet string representing the local
           forwarder attachment individual identifier (AII) to be used
           by this PW. It is used as the SAII for outgoing signaling
           messages and the TAII in the incoming messages from the
           peer."
     ::= { pwGenFecIndexMappingEntry 4 }

  pwGenFecIndexMappingRemoteAIIType OBJECT-TYPE
     SYNTAX         PwGenIdType
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "This object is the type of the the remote forwarder
           attachment individual identifier (AII) to be used
           by this PW."
     ::= { pwGenFecIndexMappingEntry 5 }

  pwGenFecIndexMappingRemoteAII OBJECT-TYPE
     SYNTAX         PwAttachmentIdentifierType
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
          "This object is an octet string representing the peer
           forwarder attachment individual identifier (AII) to be used
           by this PW. It is used as the TAII for outgoing signaling
           messages and the SAII in the incoming messages from the
           peer."
     ::= { pwGenFecIndexMappingEntry 6 }


  pwGenFecIndexMappingPwIndex  OBJECT-TYPE
```

        SYNTAX   PwIndexType
        MAX-ACCESS   read-only
        STATUS   current
        DESCRIPTION
            "The value that represents the PW in the pwTable."
        ::= { pwGenFecIndexMappingEntry 7 }

   -- End of the Gen Fec PW ID mapping table

   -- Notifications - PW

   pwDown NOTIFICATION-TYPE
       OBJECTS { pwOperStatus, --start of range
                 pwOperStatus --end of range
       }
       STATUS current
       DESCRIPTION
           "This notification is generated when the pwOperStatus
            object for one or more contiguous entries in pwTable are
            about to enter the down(2) or lowerLayerDown(6) state from
            any other state, except for transition from the
            notPresent(5) state. For the purpose of deciding when
            these notifications occur, the lowerLayerDown(6) state
            and the down(2) state are considered to be equivalent,
            i.e., there is no notification on transition from
            lowerLayerDown(6) into down(2), and there is a trap on
            transition from any other state except down(2) (and
            notPresent) into lowerLayerDown(6).

            The included values of pwOperStatus MUST all equal to this
            down(2) or lowerLayerDown(6). The two instances of
            pwOperStatus in this notification indicate the range of
            indexes that are affected. Note that all the indexes of
            the two ends of the range can be derived from the
            instance identifiers of these two objects. For cases
            where a contiguous range of cross-connects have
            transitioned into the down(2) and lowerLayerDown(6) state
            at roughly the same time, the device SHOULD issue a single
            notification for each range of contiguous indexes in an
            effort to minimize the emission of a large number of
            of notifications. If a notification has to be issued for
            just a single cross-connect entry, then the instance
            identifier (and values) of the two pwOperStatus objects
            MUST be identical."
       ::= { pwNotifications 1 }

   pwUp NOTIFICATION-TYPE
       OBJECTS { pwOperStatus, --start of range

```
                 pwOperStatus  --end of range
      }
      STATUS  current
      DESCRIPTION
          "This notification is generated when the pwOperStatus
           object for one or more contiguous entries in pwTable are
           about to enter the up(1) state from some other state
           except the notPresent(5) state and given that the pwDown
           notification been issued for these entries. The included
           values of pwOperStatus MUST both be set equal to this
           new state (i.e: up(1)).  The two instances of pwOperStatus
           in this notification indicate the range of indexes that
           are affected.  Note that all the indexes of the two ends
           of the range can be derived from the instance identifiers
           of these two objects.  For cases where a contiguous range
           of cross-connects have transitioned into the up(1) state
           at roughly the same time, the device SHOULD issue a single
           notification for each range of contiguous indexes in an
           effort to minimize the emission of a large number of
           notifications.  If a notification has to be issued for
           just a single cross-connect entry, then the instance
           identifier (and values) of the two pwOperStatus objects
           MUST be the identical."
      ::= { pwNotifications 2 }

  pwDeleted NOTIFICATION-TYPE
      OBJECTS { pwType,
                pwID,
                pwPeerAddrType,
                pwPeerAddr
      }
      STATUS  current
      DESCRIPTION
          "This notification is generated when the PW has been
           deleted, i.e. when the pwRowStatus has been set to
           destroy(6), the PW has been deleted by a non-MIB
           application or due to auto-discovery process.
           "
      ::= { pwNotifications  3 }

  -- End of notifications.

  -- Conformance information

  pwGroups      OBJECT IDENTIFIER ::= { pwConformance   1 }
  pwCompliances OBJECT IDENTIFIER ::= { pwConformance   2 }

  -- Compliance requirement for fully compliant implementations.
```

    pwModuleFullCompliance MODULE-COMPLIANCE
        STATUS   current
        DESCRIPTION
               "The compliance statement for agents that provide full
                support for PW MIB Module. Such devices can
                then be monitored and configured using
                this MIB module."

        MODULE -- this module
            MANDATORY-GROUPS { pwBasicGroup,
                               pwPerformanceGeneralGroup
                             }

      GROUP pwNotificationGroup
      DESCRIPTION "This group is only mandatory for implementations
                   which can efficiently implement the notifications
                   contained in this group.
                  "

      GROUP   pwPwIdGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support the PW ID FEC.
                  "

      GROUP   pwGeneralizedFecGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support the generalized PW FEC.
                  "

      GROUP   pwFcsGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support FCS retention."

      GROUP   pwFragGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support PW fragmentation.
                  "

      GROUP   pwPwStatusGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support PW status notification.
                  "

      GROUP   pwGetNextGroup
      DESCRIPTION "This group is only mandatory for implementations
                   where the pwIndex may be any arbitrary value
                   and the EMS would require retrieval of the next
                   free index."

```
     GROUP          pwPriorityGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support the controlling the PW setup and
                  holding priority."

     GROUP          pwAttachmentGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support attachment of two PWs (PW stitching)."

     GROUP          pwPeformance1DayIntervalGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support PW performance gathering in 1 day
                  intervals."

     GROUP          pwPerformanceIntervalGeneralGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support PW performance gathering in 15
                  minute intervals."

     GROUP          pwPeformanceIntervalGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support PW performance gathering in 15
                  minute intervals."

     GROUP          pwHCPeformanceIntervalGroup
     DESCRIPTION "This group is only mandatory for implementations
                  where at least one of the interval performance
                  counters wraps around too quickly based on the
                  criteria specified in RFC 2863 for high-capacity
                  counters."

     GROUP          pwMappingTablesGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support reverse mapping of PW indexes to
                  the pwIndex and the peer mapping table."

     GROUP          pwSignalingGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support the PW signaling."

     GROUP          pwNotificationControlGroup
     DESCRIPTION "This group is only mandatory for implementations
                  that support the PW notifications."

     OBJECT         pwAdminStatus
     SYNTAX         INTEGER { up(1), down(2) }
     DESCRIPTION "The support of the value testing(3) is not
                  required."
```

```
     OBJECT          pwOperStatus
     SYNTAX          INTEGER { up(1), down(2), notPresent(5),
                     lowerLayerDown(6) }
     DESCRIPTION "The support of the values testing(3) and dormant(4)
                     is not required."


     OBJECT          pwRowStatus
     SYNTAX          RowStatus { active(1), notInService(2),
                                 notReady(3) }
     WRITE-SYNTAX RowStatus { active(1), notInService(2),
                                 createAndGo(4), destroy(6)
                                 }
     DESCRIPTION "Support for createAndWait is not required. Support
                     of notReady is not required for implementations
                     that do not support signaling, or if it is
                     guaranteed that the conceptual row has all the
                     required information to create the PW when the
                     row has been created by the agent or written by
                     the operator."


     OBJECT          pwPeerAddrType
     SYNTAX          InetAddressType { unknown(0), ipv4(1) }
     MIN-ACCESS   read-only
     DESCRIPTION "Only unknown(0) and ipv4(1) is required.
                     Implementation that support only IPv4 MAY support
                     read-only access."


     OBJECT          pwPeerAddr
     SYNTAX          InetAddress (SIZE(0|4))
     DESCRIPTION "An implementation is only required to support
                     0, 4 address sizes."


     OBJECT          pwStorageType
     MIN-ACCESS   read-only
     DESCRIPTION "Write access is not required."


     OBJECT          pwNotifRate
     MIN-ACCESS   read-only
     DESCRIPTION "Write access is not required."


        ::= { pwCompliances 1 }

  -- Compliance requirement for read-only compliant implementations.

  pwModuleReadOnlyCompliance MODULE-COMPLIANCE
     STATUS   current
     DESCRIPTION
             "The compliance statement for agents that provide read-
```

                    only support for PW MIB Module. Such devices can
                    then be monitored but cannot be configured using this
                    MIB module."

         MODULE -- this module
             MANDATORY-GROUPS { pwBasicGroup
                               }

         GROUP   pwNotificationGroup
         DESCRIPTION "This group is only mandatory for implementations
                       which can efficiently implement the notifications
                       contained in this group."

         GROUP   pwPwIdGroup
         DESCRIPTION "This group is only mandatory for implementations
                       that support the PW ID FEC.
                      "

         GROUP   pwGeneralizedFecGroup
         DESCRIPTION "This group is only mandatory for implementations
                       that support the generalized PW FEC.
                      "

         GROUP   pwFcsGroup
         DESCRIPTION "This group is only mandatory for implementations
                       that support FCS retention."

         GROUP   pwFragGroup
         DESCRIPTION "This group is only mandatory for implementations
                       that support PW fragmentation.
                      "

         GROUP   pwPwStatusGroup
         DESCRIPTION "This group is only mandatory for implementations
                       that support PW status notification.
                      "

         GROUP   pwGetNextGroup
         DESCRIPTION "This group is only mandatory for implementations
                       where the pwIndex may be any arbitrary value
                       and the EMS would require retrieval of the next
                       free index."

         GROUP   pwPriorityGroup
         DESCRIPTION "This group is only mandatory for implementations
                       that support the controlling the PW setup and
                       holding priority."

```
      GROUP          pwAttachmentGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support attachment of two PWs (PW stitching)."

      GROUP          pwPeformance1DayIntervalGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support PW performance gathering in 1 Day
                   intervals."

      GROUP          pwPerformanceIntervalGeneralGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support PW performance gathering in 15
                   minute intervals."

      GROUP          pwPeformanceIntervalGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support PW performance gathering in 15
                   minute intervals."

      GROUP          pwHCPeformanceIntervalGroup
      DESCRIPTION "This group is only mandatory for implementations
                   where at least one of the interval performance
                   counters wraps around too quickly based on the
                   criteria specified in RFC 2863 for high-capacity
                   counters."

      GROUP          pwMappingTablesGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support reverse mapping of PW indexes to
                   the pwIndex and the peer mapping table."

      GROUP          pwSignalingGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support the PW signaling."

      GROUP          pwNotificationControlGroup
      DESCRIPTION "This group is only mandatory for implementations
                   that support the PW notifications."

      OBJECT         pwType
      MIN-ACCESS   read-only
      DESCRIPTION "Write access is not required."

      OBJECT         pwOwner
      MIN-ACCESS   read-only
      DESCRIPTION "Write access is not required."

      OBJECT         pwPsnType
```

```
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwSetUpPriority
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwHoldingPriority
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwPeerAddrType
    SYNTAX       InetAddressType { unknown(0), ipv4(1) }
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required. Only unknown(0) and
                 ipv4(1) is required."

    OBJECT       pwPeerAddr
    SYNTAX       InetAddress (SIZE(0|4))
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required. An implementation
                 is only required to support 0, 4 address sizes."

    OBJECT       pwAttachedPwIndex
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwIfIndex
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwID
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwLocalGroupID
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwGroupAttachmentID
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwLocalAttachmentID
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT       pwRemoteAttachmentID
```

```
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwCwPreference
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwLocalIfMtu
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwLocalIfString
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwLocalCapabAdvert
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwFragmentCfgSize
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwFcsRetentionCfg
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwOutboundLabel
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwInboundLabel
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwName
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwDescr
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required."

        OBJECT       pwAdminStatus
        SYNTAX       INTEGER { up(1), down(2) }
        MIN-ACCESS   read-only
        DESCRIPTION "Write access is not required. The support of value
                     testing(3) is not required."
```

```
    OBJECT        pwOperStatus
    SYNTAX        INTEGER { up(1), down(2), notPresent(5),
                  lowerLayerDown(6) }
    DESCRIPTION "The support of the values testing(3) and dormant(4)
                  is not required."

    OBJECT        pwRowStatus
    SYNTAX        RowStatus { active(1) }
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwStorageType
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwOamEnable
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwGenAGIType
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwGenLocalAIIType
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwGenRemoteAIIType
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwUpDownNotifEnable
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwDeletedNotifEnable
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

    OBJECT        pwNotifRate
    MIN-ACCESS   read-only
    DESCRIPTION "Write access is not required."

        ::= { pwCompliances 2 }

  -- Units of conformance.

  pwBasicGroup    OBJECT-GROUP
```

```
     OBJECTS {
              pwType,
              pwOwner,
              pwPsnType,
              pwIfIndex,
              pwCwPreference,
              pwLocalIfMtu,
              pwOutboundLabel,
              pwInboundLabel,
              pwName,
              pwDescr,
              pwCreateTime,
              pwUpTime,
              pwLastChange,
              pwAdminStatus,
              pwOperStatus,
              pwLocalStatus,
              pwRowStatus,
              pwStorageType,
              pwOamEnable
            }

     STATUS  current
     DESCRIPTION
         "Collection of objects that are required in all
          implementations that support the PW MIB module."
     ::= { pwGroups 1 }

  pwPwIdGroup   OBJECT-GROUP
     OBJECTS {
              pwID
             }

     STATUS  current
     DESCRIPTION
         "Collection of objects required for PW ID configuration
          and signaling."
     ::= { pwGroups 2 }

  pwGeneralizedFecGroup   OBJECT-GROUP
     OBJECTS {
              pwGroupAttachmentID,
              pwLocalAttachmentID,
              pwRemoteAttachmentID,
              pwGenAGIType,
              pwGenLocalAIIType,
              pwGenRemoteAIIType
            }
```

```
     STATUS  current
     DESCRIPTION
         "Collection of objects required for generalized FEC
          configuration and signaling."
     ::= { pwGroups 3 }

  pwFcsGroup   OBJECT-GROUP
     OBJECTS {
               pwFcsRetentionCfg,
               pwFcsRetentionStatus
             }

     STATUS  current
     DESCRIPTION
         "Collection of objects required for FCS retention
          configuration and signaling."
     ::= { pwGroups 4 }

  pwFragGroup   OBJECT-GROUP
     OBJECTS {
               pwFragmentCfgSize,
               pwRmtFragCapability
             }

     STATUS  current
     DESCRIPTION
         "Collection of objects required for fragmentation
          configuration and signaling."
     ::= { pwGroups 5 }

  pwPwStatusGroup   OBJECT-GROUP
     OBJECTS {
               pwRemoteCapabilities,
               pwRemoteStatusCapable,
               pwRemoteStatus
             }

     STATUS  current
     DESCRIPTION
         "Collection of objects required for PW status configuration
          and signaling."
     ::= { pwGroups 6 }


  pwGetNextGroup    OBJECT-GROUP
     OBJECTS {
               pwIndexNext
               }
```

```
        STATUS  current
        DESCRIPTION
            "Collection of objects for getting the next available
             index."
        ::= { pwGroups 7 }

    pwPriorityGroup   OBJECT-GROUP
        OBJECTS {
                 pwSetUpPriority,
                 pwHoldingPriority
                 }

        STATUS  current
        DESCRIPTION
            "Collection of objects for controlling the PW setup and
             holding priority."
        ::= { pwGroups 8 }

    pwAttachmentGroup   OBJECT-GROUP
        OBJECTS {
                 pwAttachedPwIndex
                 }

        STATUS  current
        DESCRIPTION
            "Collection of objects for PW configuration as ifIndex"
        ::= { pwGroups 9 }

    pwPerformanceGeneralGroup OBJECT-GROUP
        OBJECTS {
                 pwPerfTotalErrorPackets
               }

        STATUS  current
        DESCRIPTION
            "Collection of general objects needed for managing the
             total running performance parameters."
        ::= { pwGroups 10 }

    pwPeformance1DayIntervalGroup OBJECT-GROUP
        OBJECTS {
                 pwPerf1DayIntervalValidData,
                 pwPerf1DayIntervalTimeElapsed,
                 pwPerf1DayIntervalInHCPackets,
                 pwPerf1DayIntervalInHCBytes,
                 pwPerf1DayIntervalOutHCPackets,
                 pwPerf1DayIntervalOutHCBytes
               }
```

        STATUS current
        DESCRIPTION
            "Collection of objects needed for PW running 1 day interval
             performance collection."
        ::= { pwGroups 11 }

   pwPerformanceIntervalGeneralGroup OBJECT-GROUP
       OBJECTS {
                pwTimeElapsed,
                pwValidIntervals,
                pwPerfIntervalValidData,
                pwPerfIntervalTimeElapsed
              }

       STATUS current
       DESCRIPTION
           "Collection of general objects needed for managing the
            interval performance parameters."
       ::= { pwGroups 12 }

   pwPeformanceIntervalGroup OBJECT-GROUP
       OBJECTS {
                pwPerfCurrentInPackets,
                pwPerfCurrentInBytes,
                pwPerfCurrentOutPackets,
                pwPerfCurrentOutBytes,

                pwPerfIntervalInPackets,
                pwPerfIntervalInBytes,
                pwPerfIntervalOutPackets,
                pwPerfIntervalOutBytes
              }

       STATUS current
       DESCRIPTION
           "Collection of 32 bits objects needed for PW performance
            collection in 15 minutes intervals."
       ::= { pwGroups 13 }

   pwHCPeformanceIntervalGroup OBJECT-GROUP
       OBJECTS {
                pwPerfCurrentInHCPackets,
                pwPerfCurrentInHCBytes,
                pwPerfCurrentOutHCPackets,
                pwPerfCurrentOutHCBytes,

                pwPerfIntervalInHCPackets,
                pwPerfIntervalInHCBytes,

```
                pwPerfIntervalOutHCPackets,
                pwPerfIntervalOutHCBytes
              }

      STATUS  current
      DESCRIPTION
          "Collection of HC objects needed for PW performance
           collection in 15 minutes intervals."
      ::= { pwGroups 14 }

   pwMappingTablesGroup OBJECT-GROUP
      OBJECTS {
                pwIndexMappingPwIndex,
                pwPeerMappingPwIndex,
                pwGenFecIndexMappingPwIndex
              }

      STATUS  current
      DESCRIPTION
          "Collection of objects contained in the reverse
           mapping tables."
      ::= { pwGroups 15 }

   pwNotificationControlGroup OBJECT-GROUP
      OBJECTS {
                pwUpDownNotifEnable,
                pwDeletedNotifEnable,
                pwNotifRate
              }

      STATUS  current
      DESCRIPTION
          "Collection of objects for controlling the PW
           notifications."
      ::= { pwGroups 16 }

   pwNotificationGroup NOTIFICATION-GROUP
      NOTIFICATIONS {
                pwUp,
                pwDown,
                pwDeleted
              }

      STATUS  current
      DESCRIPTION
          "Collection PW notifications objects."
      ::= { pwGroups 17 }
```

```
  pwSignalingGroup OBJECT-GROUP
     OBJECTS {
               pwPeerAddrType,
               pwPeerAddr,
               pwLocalGroupID,
               pwLocalIfString,
               pwLocalCapabAdvert,
               pwRemoteGroupID,
               pwCwStatus,
               pwRemoteIfMtu,
               pwRemoteIfString
             }

     STATUS  current
     DESCRIPTION
         "Collection of objects for use in implementations that
          support the PW signaling."
     ::= { pwGroups 18 }

  END
```

14. Security Considerations

   It is clear that this MIB module is potentially useful for monitoring
   PW capable PEs. This MIB module can also be used for configuration
   of certain objects, and anything that can be configured can be
   incorrectly configured, with potentially disastrous results.

   There are number of management objects defined in this MIB module
   with a MAX-ACCESS clause of read-write and/or read-create. Such
   objects may be considered sensitive or vulnerable in some network
   environments. The support for SET operations in a non-secure
   environment without proper protection can have a negative effect on
   network operations. These are the tables and objects and their
   sensitivity/vulnerability:

   o the pwTable contains objects to configure PW parameters on a
      Provider Edge (PE) device. Unauthorized access to objects in this
      table, could result in disruption of traffic on the network. The
      objects pwUpDownNotifEnable and pwNotifRate control the reports
      from the network element to the EMS. Unauthorized access to these
      objects could result in disruption of configuration and status
      change reporting, resulting mis-view of the network conditions.
      The use of stronger mechanisms such as SNMPv3 security should be

considered where possible. Specifically, SNMPv3 VACM and USM MUST
be used with any v3 agent which implements this MIB module.
Administrators should consider whether read access to these
objects should be allowed, since read access may be undesirable
under certain circumstances.

Some of the readable objects in this MIB module (i.e., objects with a
MAX-ACCESS other than not-accessible) may be considered sensitive or
vulnerable in some network environments. It is thus important to
control even GET and/or NOTIFY access to these objects and possibly
to even encrypt the values of these objects when sending them over
the network via SNMP. These are the tables and objects and their
sensitivity/vulnerability:

o   the pwTable, pwPerfCurrentTable, pwPerfIntervalTable,
    pwPerf1DayIntervalTable, pwIndexMappingTable, pwPeerMappingTable
    and pwGenFecIndexMappingTable collectively show the pseudowire
    connectivity topology and its performance characteristics. If an
    Administrator does not want to reveal this information, then these
    tables should be considered sensitive/vulnerable.

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 con8figured to give access
to the objects only to those principals (users) that have legitimate
rights to indeed GET or SET (change/create/delete) them.

15. IANA Considerations

15.1. ifType for PW

IANA has beeen requested to register a value for PW in the
IANAifType-MIB called ifPwType. When the assignment has been made,
the RFC Editor is asked to document the value here.

15.2. PW MIB Modules OBJECT IDENTIFIER values

A PW may appear as ifIndex in the ifTable, and therefore it is
requested below that the pwStdMIB OBJECT IDENTIFIER will be assigned
under the 'transmission' subtree, as the common practice in assigning
OBJECT IDENTIFIERs for MIB modules representing entities in the
ifTable.

All other MIB modules related to PW management SHOULD be assigned
under the 'mib-2' subtree, individual request will appear in the MIB
module memo IANA Considerations section.

15.3. IANA Considerations for PW-STD-MIB

The MIB module in this document uses the following IANA-assigned
OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

```
     Descriptor         OBJECT IDENTIFIER value
     ----------         -----------------------

     pwStdMIB           { transmission ZZZZ }
```

Editor's Note (to be removed prior to publication): The IANA is
requested to assign a value for "ZZZZ" under the 'transmission'
subtree and to record the assignment in the SMI Numbers registry.
This value SHOULD be the same value assigned for the PW ifType for
representation of the PW in the ifTable. When the assignment has
been made, the RFC Editor is asked to replace "ZZZZ" (here and in the
MIB module) with the assigned value and to remove this note.

15.4. IANA Considerations for IANA-PWE3-MIB

The MIB module in this document uses the following IANA-assigned
OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

```
     Descriptor         OBJECT IDENTIFIER value
     ----------         -----------------------

     ianaPwe3MIB        { mib-2 XXXX }
```

Editor's Note (to be removed prior to publication): The IANA is
requested to assign a value for "XXXX" under the 'mib-2' subtree and
to record the assignment in the SMI Numbers registry. When the
assignment has been made, the RFC Editor is asked to replace "XXXX"
(here and in the MIB module) with the assigned value and to remove
this note.

16. Acknowledgements

   We thank Orly Nicklass for her dedicated review and significant edit
   at various sections of the document, and Kiran Koushik for his
   contribution.

17. References

17.1. Normative References

   [BCP14]   Bradner, S., "Key words for use in RFCs to Indicate
               requirement Levels", BCP 14, RFC 2119, March 1997.

   [PWTC]   Nadeau, T. and D. Zelig, "Definitions for Textual
               Conventions and OBJECT-IDENTITIES for Pseudowires
               Management", work-in-progress .

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

   [RFC2579]   McCloghrie, K., Ed., Perkins, D., Ed., and J.
               Schoenwaelder, Ed., "Textual Conventions for SMIv2",
               STD 58, RFC 2579, April 1999.

   [RFC2580]   McCloghrie, K., Perkins, D., and J. Schoenwaelder,
               "Conformance Statements for SMIv2", STD 58, RFC 2580,
               April 1999.

   [RFC2863]   McCloghrie, K. and F. Kastenholz, "The Interfaces Group
               MIB", RFC 2863, June 2000.

   [RFC3411]   Harrington, D., Presuhn, R., and B. Wijnen, "An
               Architecture for Describing Simple Network Management
               Protocol (SNMP) Management Frameworks", STD 62, RFC 3411,
               December 2002.

   [RFC3413]   Levi, D., Meyer, P., and B. Stewart, "Simple Network
               Management Protocol (SNMP) Applications", STD 62,
               RFC 3413, December 2002.

   [RFC3593]   Tesink, K., "Textual Conventions for MIB Modules Using
               Performance History Based on 15 Minute Intervals",
               RFC 3593, September 2003.

   [RFC3705]   Ray, B. and R. Abbi, "High Capacity Textual Conventions
               for MIB Modules Using Performance History Based on 15

Minute Intervals", RFC 3705, February 2004.

[RFC3931]   Lau, J., Townsley, M., and I. Goyret, "Layer Two Tunneling
            Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.

[RFC4001]   Daniele, M., Haberman, B., Routhier, S., and J.
            Schoenwaelder, "Textual Conventions for Internet Network
            Addresses", RFC 4001, February 2005.

[RFC4446]   Martini, L., "IANA Allocations for Pseudowire Edge to Edge
            Emulation (PWE3)", BCP 116, RFC 4446, April 2006.

[RFC4447]   Martini, L., Rosen, E., El-Aawar, N., Smith, T., and G.
            Heron, "Pseudowire Setup and Maintenance Using the Label
            Distribution Protocol (LDP)", RFC 4447, April 2006.

[RFC4623]   Malis, A. and M. Townsley, "Pseudowire Emulation Edge-to-
            Edge (PWE3) Fragmentation and Reassembly", RFC 4623,
            August 2006.

[RFC4720]   Malis, A., Allan, D., and N. Del Regno, "Pseudowire
            Emulation Edge-to-Edge (PWE3) Frame Check Sequence
            Retention", RFC 4720, November 2006.

[RFC4863]   Martini, L. and G. Swallow, "Wildcard Pseudowire Type",
            RFC 4863, May 2007.

17.2. Informative References

[CEPMIB]   Zelig, D., Nadeau, T., and R. Cohen, "SONET/SDH Circuit
            Emulation Service Over Packet (CEP) Management Information
            Base Using SMIv2", work-in-progress .

[PWMPLSMIB]
            Zelig, D. and T. Nadeau, "Pseudowire (PW) Over MPLS PSN
            Management Information Base", work-in-progress .

[RFC2434]   Narten, T. and H. Alvestrand, "Guidelines for Writing an
            IANA Considerations Section in RFCs", BCP 26, RFC 2434,
            October 1998.

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

[RFC3916]   Xiao, X., McPherson, D., and P. Pate, "Requirements for
            Pseudo-Wire Emulation Edge-to-Edge (PWE3)", RFC 3916,
            September 2004.

   [RFC3985]   Bryant, S. and P. Pate, "Pseudo Wire Emulation Edge-to-
               Edge (PWE3) Architecture", RFC 3985, March 2005.

Authors' Addresses

   Thomas D. Nadeau (editor)
   BT
   BT Centre
   81 Newgate Street
   London EC1A 7AJ
   United Kingdom

   Email: thomas.nadeau@bt.com

   David Zelig (editor)
   Corrigent Systems
   126, Yigal Alon St.
   Tel Aviv,
   Israel

   Phone: +972 3 6945 273
   Email: davidz@corrigent.com

Full Copyright Statement

Intellectual Property

Acknowledgment