              Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture

Status of This Memo

Copyright Notice

Abstract

   This document describes an architecture for Pseudo Wire Emulation
   Edge-to-Edge (PWE3). It discusses the emulation of services such as
   Frame Relay, ATM, Ethernet, TDM, and SONET/SDH over packet switched
   networks (PSNs) using IP or MPLS. It presents the architectural
   framework for pseudo wires (PWs), defines terminology, and specifies
   the various protocol elements and their functions.

Table of Contents

1. Introduction

   This document describes an architecture for Pseudo Wire Emulation
   Edge-to-Edge (PWE3) in support of [RFC3916]. It discusses the
   emulation of services such as Frame Relay, ATM, Ethernet, TDM, and
   SONET/SDH over packet switched networks (PSNs) using IP or MPLS. It
   presents the architectural framework for pseudo wires (PWs), defines
   terminology, and specifies the various protocol elements and their
   functions.

1.1. Pseudo Wire Definition

   PWE3 is a mechanism that emulates the essential attributes of a
   telecommunications service (such as a T1 leased line or Frame Relay)
   over a PSN. PWE3 is intended to provide only the minimum necessary
   functionality to emulate the wire with the required degree of
   faithfulness for the given service definition. Any required
   switching functionality is the responsibility of a forwarder function

(FWRD). Any translation or other operation needing knowledge of the
payload semantics is carried out by native service processing (NSP)
elements. The functional definition of any FWRD or NSP elements is
outside the scope of PWE3.

The required functions of PWs include encapsulating service-specific
bit streams, cells, or PDUs arriving at an ingress port and carrying
them across an IP path or MPLS tunnel. In some cases it is necessary
to perform other operations such as managing their timing and order,
to emulate the behavior and characteristics of the service to the
required degree of faithfulness.

From the perspective of Customer Edge Equipment (CE), the PW is
characterized as an unshared link or circuit of the chosen service.
In some cases, there may be deficiencies in the PW emulation that
impact the traffic carried over a PW and therefore limit the
applicability of this technology. These limitations must be fully
described in the appropriate service-specific documentation.

For each service type, there will be one default mode of operation
that all PEs offering that service type must support. However,
optional modes may be defined to improve the faithfulness of the
emulated service, if it can be clearly demonstrated that the
additional complexity associated with the optional mode is offset by
the value it offers to PW users.

1.2. PW Service Functionality

PWs provide the following functions in order to emulate the behavior
and characteristics of the native service.

   o Encapsulation of service-specific PDUs or circuit data arriving
     at the PE-bound port (logical or physical).
   o Carriage of the encapsulated data across a PSN tunnel.
   o Establishment of the PW, including the exchange and/or
     distribution of the PW identifiers used by the PSN tunnel
     endpoints.
   o Managing the signaling, timing, order, or other aspects of the
     service at the boundaries of the PW.
   o Service-specific status and alarm management.

1.3. Non-Goals of This Document

   The following are non-goals for this document:

      o The on-the-wire specification of PW encapsulations.
      o The detailed definition of the protocols involved in PW setup
         and maintenance.

   The following are outside the scope of PWE3:

      o Any multicast service not native to the emulated medium. Thus,
         Ethernet transmission to a "multicast" IEEE-48 address is in
         scope, but multicast services such as MARS [RFC2022] that are
         implemented on top of the medium are not.
      o Methods to signal or control the underlying PSN.

1.4. Terminology

   This document uses the following definitions of terms. These terms
   are illustrated in context in Figure 2.

   Attachment Circuit   The physical or virtual circuit attaching
   (AC)   a CE to a PE. An attachment Circuit may be, for
                         example, a Frame Relay DLCI, an ATM VPI/VCI, an
                         Ethernet port, a VLAN, a PPP connection on a
                         physical interface, a PPP session from an L2TP
                         tunnel, or an MPLS LSP. If both physical and
                         virtual ACs are of the same technology (e.g.,
                         both ATM, both Ethernet, both Frame Relay), the
                         PW is said to provide "homogeneous transport";
                         otherwise, it is said to provide "heterogeneous
                         transport".

   CE-bound   The traffic direction in which PW-PDUs are
                         received on a PW via the PSN, processed, and
                         then sent to the destination CE.

   CE Signaling   Messages sent and received by the CE's control
                         plane. It may be desirable or even necessary
                         for the PE to participate in or to monitor this
                         signaling in order to emulate the service
                         effectively.

   Control Word (CW)   A four-octet header used in some encapsulations
                         to carry per-packet information when the PSN is
                         MPLS.

Customer Edge (CE)   A device where one end of a service originates
                      and/or terminates. The CE is not aware that it
                      is using an emulated service rather than a
                      native service.

Forwarder (FWRD)   A PE subsystem that selects the PW to use in
                      order to transmit a payload received on an AC.

Fragmentation   The action of dividing a single PDU into
                      multiple PDUs before transmission with the
                      intent of the original PDU being reassembled
                      elsewhere in the network. Packets may undergo
                      fragmentation if they are larger than the MTU of
                      the network they will traverse.

Maximum Transmission The packet size (excluding data link header)
unit (MTU)   that an interface can transmit without needing
                      to fragment.

Native Service   Processing of the data received by the PE
Processing (NSP)   from the CE before presentation to the PW for
                      transmission across the core, or processing of
                      the data received from a PW by a PE before it is
                      output on the AC. NSP functionality is defined
                      by standards bodies other than the IETF, such as
                      ITU-T,ANSI, or ATMF.)

Packet Switched   Within the context of PWE3, this is a
Network (PSN)   network using IP or MPLS as the mechanism for
                      packet forwarding.

PE-Bound   The traffic direction in which information from
                      a CE is adapted to a PW, and PW-PDUs are sent
                      into the PSN.

PE/PW Maintenance   Used by the PEs to set up, maintain, and tear
                      down the PW. It may be coupled with CE
                      Signaling in order to manage the PW effectively.

Protocol Data   The unit of data output to, or received
Unit (PDU)   from, the network by a protocol layer.

Provider Edge (PE)   A device that provides PWE3 to a CE.

Pseudo Wire (PW)   A mechanism that carries the essential elements
                      of an emulated service from one PE to one or
                      more other PEs over a PSN.

Pseudo Wire   A mechanism that emulates the essential
Emulation Edge to   attributes of service (such as a T1 leased
Edge (PWE3)   line or Frame Relay) over a PSN.

Pseudo Wire PDU   A PDU sent on the PW that contains all of
(PW-PDU)   the data and control information necessary to
                     emulate the desired service.

PSN Tunnel   A tunnel across a PSN, inside which one or more
                     PWs can be carried.

PSN Tunnel   Used to set up, maintain, and tear down the
Signaling   underlying PSN tunnel.

PW Demultiplexer   Data-plane method of identifying a PW
                     terminating at a PE.

Time Domain   Time Division Multiplexing. Frequently used
Multiplexing (TDM)   to refer to the synchronous bit streams at rates
                     defined by G.702.

Tunnel   A method of transparently carrying information
                     over a network.

2. PWE3 Applicability

   The PSN carrying a PW will subject payload packets to loss, delay,
   delay variation, and re-ordering. During a network transient there
   may be a sustained period of impaired service. The applicability of
   PWE3 to a particular service depends on the sensitivity of that
   service (or the CE implementation) to these effects, and on the
   ability of the adaptation layer to mask them. Some services, such as
   IP over FR over PWE3, may prove quite resilient to IP and MPLS PSN
   characteristics. Other services, such as the interconnection of PBX
   systems via PWE3, will require more careful consideration of the PSN
   and adaptation layer characteristics. In some instances, traffic
   engineering of the underlying PSN will be required, and in some cases
   the constraints may make the required service guarantees impossible
   to provide.

3. Protocol Layering Model

   The PWE3 protocol-layering model is intended to minimize the
   differences between PWs operating over different PSN types. The
   design of the protocol-layering model has the goals of making each PW
   definition independent of the underlying PSN, and of maximizing the
   reuse of IETF protocol definitions and their implementations.

3.1. Protocol Layers

   The logical protocol-layering model required to support a PW is shown
   in Figure 1.

```
        +--------------------------+
        |         Payload          |
        +--------------------------+
        |       Encapsulation      | <==== may be empty
        +--------------------------+
        |     PW Demultiplexer     |
        +--------------------------+
        |      PSN Convergence     | <==== may be empty
        +--------------------------+
        |           PSN            |
        +--------------------------+
        |        Data-Link         |
        +--------------------------+
        |         Physical         |
        +--------------------------+
```
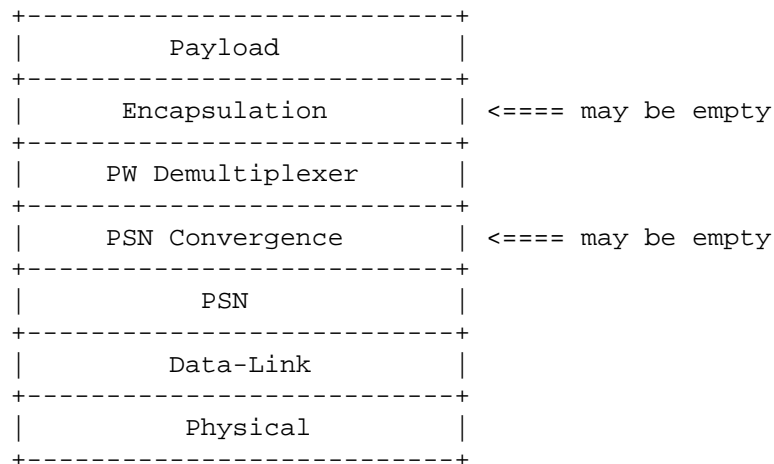
                 Figure 1. Logical Protocol Layering Model

   The payload is transported over the Encapsulation Layer. The
   Encapsulation Layer carries any information, not already present
   within the payload itself, that is needed by the PW CE-bound PE
   interface to send the payload to the CE via the physical interface.
   If no further information is needed in the payload itself, this layer
   is empty.

   The Encapsulation Layer also provides support for real-time
   processing, and if needed for sequencing.

   The PW Demultiplexer layer provides the ability to deliver multiple
   PWs over a single PSN tunnel. The PW demultiplexer value used to
   identify the PW in the data plane may be unique per PE, but this is
   not a PWE3 requirement. It must, however, be unique per tunnel
   endpoint. If it is necessary to identify a particular tunnel, then
   that is the responsibility of the PSN layer.

   The PSN Convergence layer provides the enhancements needed to make
   the PSN conform to the assumed PSN service requirement. Therefore,
   this layer provides a consistent interface to the PW, making the PW
   independent of the PSN type. If the PSN already meets the service
   requirements, this layer is empty.

The PSN header, MAC/Data-Link, and Physical Layer definitions are
outside the scope of this document. The PSN can be IPv4, IPv6, or
MPLS.

3.2. Domain of PWE3

PWE3 defines the Encapsulation Layer, the method of carrying various
payload types, and the interface to the PW Demultiplexer Layer. It
is expected that the other layers will be provided by tunneling
methods such as L2TP or MPLS over the PSN.

3.3. Payload Types

The payload is classified into the following generic types of native
data units:

        o Packet
        o Cell
        o Bit stream
        o Structured bit stream

Within these generic types there are specific service types:

     Generic Payload Type   PW Service
     --------------------   ----------
     Packet   Ethernet (all types), HDLC framing,
                              Frame Relay, ATM AAL5 PDU.

     Cell   ATM.

     Bit stream   Unstructured E1, T1, E3, T3.

     Structured bit stream   SONET/SDH (e.g., SPE, VT, NxDS0).

3.3.1. Packet Payload

A packet payload is a variable-size data unit delivered to the PE via
the AC. A packet payload may be large compared to the PSN MTU. The
delineation of the packet boundaries is encapsulation specific. HDLC
or Ethernet PDUs can be considered examples of packet payloads.
Typically, a packet will be stripped of transmission overhead such as
HDLC flags and stuffing bits before transmission over the PW.

A packet payload would normally be relayed across the PW as a single
unit. However, there will be cases where the combined size of the
packet payload and its associated PWE3 and PSN headers exceeds the
PSN path MTU. In these cases, some fragmentation methodology has to
be applied. This may, for example, be the case when a user provides

the service and attaches to the service provider via Ethernet, or
when nested pseudo-wires are involved. Fragmentation is discussed in
more detail in section 5.3.

A packet payload may need sequencing and real-time support.

In some situations, the packet payload may be selected from the
packets presented on the emulated wire on the basis of some sub-
multiplexing technique. For example, one or more Frame Relay PDUs
may be selected for transport over a particular pseudo wire based on
the Frame Relay Data-Link Connection Identifier (DLCI), or, in the
case of Ethernet payloads, by using a suitable MAC bridge filter.
This is a forwarder function, and this selection would therefore be
made before the packet was presented to the PW Encapsulation Layer.

3.3.2. Cell Payload

A cell payload is created by capturing, transporting, and replaying
groups of octets presented on the wire in a fixed-size format. The
delineation of the group of bits that comprise the cell is specific
to the encapsulation type. Two common examples of cell payloads are
ATM 53-octet cells, and the larger 188-octet MPEG Transport Stream
packets [DVB].

To reduce per-PSN packet overhead, multiple cells may be concatenated
into a single payload. The Encapsulation Layer may consider the
payload complete on the expiry of a timer, after a fixed number of
cells have been received or when a significant cell (e.g., an ATM OAM
cell) has been received. The benefit of concatenating multiple PDUs
should be weighed against a possible increase in packet delay
variation and the larger penalty incurred by packet loss. In some
cases, it may be appropriate for the Encapsulation Layer to perform
some type of compression, such as silence suppression or voice
compression.

The generic cell payload service will normally need sequence number
support and may also need real-time support. The generic cell
payload service would not normally require fragmentation.

The Encapsulation Layer may apply some form of compression to some of
these sub-types (e.g., idle cells may be suppressed).

In some instances, the cells to be incorporated in the payload may be
selected by filtering them from the stream of cells presented on the
wire. For example, an ATM PWE3 service may select cells based on
their VCI or VPI fields. This is a forwarder function, and the
selection would therefore be made before the packet was presented to
the PW Encapsulation Layer.

3.3.3. Bit Stream

   A bit stream payload is created by capturing, transporting, and
   replaying the bit pattern on the emulated wire, without taking
   advantage of any structure that, on inspection, may be visible within
   the relayed traffic (i.e., the internal structure has no effect on
   the fragmentation into packets).

   In some instances it is possible to apply suppression to bit streams.
   For example, E1 and T1 send "all-ones" to indicate failure. This
   condition can be detected without any knowledge of the structure of
   the bit stream, and transmission of packetized can be data
   suppressed.

   This service will require sequencing and real-time support.

3.3.4. Structured Bit Stream

   A structured bit stream payload is created by using some knowledge of
   the underlying structure of the bit stream to capture, transport, and
   replay the bit pattern on the emulated wire.

   Two important points distinguish structured and unstructured bit
   streams:

      o Some parts of the original bit stream may be stripped in the
        PSN-bound direction by an NSP block. For example, in
        Structured SONET the section and line overhead (and possibly
        more) may be stripped. A framer is required to enable such
        stripping. It is also required for frame/payload alignment for
        fractional T1/E1 applications.

      o The PW must preserve the structure across the PSN so that the
        CE-bound NSP block can insert it correctly into the
        reconstructed unstructured bit stream. The stripped
        information (such as SONET pointer justifications) may appear
        in the encapsulation layer to facilitate this reconstitution.

   As an option, the Encapsulation Layer may also perform silence/idle
   suppression or similar compression on a structured bit stream.

   Structured bit streams are distinguished from cells in that the
   structures may be too long to be carried in a single packet. Note
   that "short" structures are indistinguishable from cells and may
   benefit from the use of methods described in section 3.3.2.

   This service requires sequencing and real-time support.

3.3.5. Principle of Minimum Intervention

   To minimize the scope of information, and to improve the efficiency
   of data flow through the Encapsulation Layer, the payload should be
   transported as received, with as few modifications as possible
   [RFC1958].

   This minimum intervention approach decouples payload development from
   PW development and requires fewer translations at the NSP in a system
   with similar CE interfaces at each end. It also prevents unwanted
   side effects due to subtle misrepresentation of the payload in the
   intermediate format.

   An approach that does intervene can be more wire efficient in some
   cases and may result in fewer translations at the NSP whereby the CE
   interfaces are of different types. Any intermediate format
   effectively becomes a new framing type, requiring documentation and
   assured interoperability. This increases the amount of work for
   handling the protocol that the intermediate format carries and is
   undesirable.

4. Architecture of Pseudo Wires

   This section describes the PWE3 architectural model.

4.1. Network Reference Model

   Figure 2 illustrates the network reference model for point-to-point
   PWs.

```
          |<-------------- Emulated Service --------------->|
          |                                                 |
          |            |<------- Pseudo Wire ------>|        |
          |            |                            |        |
          |            |     |<-- PSN Tunnel -->|    |        |
          |            V     V                  V    V        |
          V    AC    +----+                    +----+   AC   V
   +-----+    |    | PE1|================| PE2|    |    +-----+
   |     |----------|............PW1.............|----------|     |
   | CE1 |    |    |    |                         |    |    | CE2 |
   |     |----------|............PW2.............|----------|     |
   +-----+  ^ |    |    |================|    |    | ^  +-----+
        ^   |    +----+                    +----+    | |  ^
        |   |    Provider Edge 1      Provider Edge 2  | |
        |   |                                          | |
   Customer |                                        | Customer
   Edge 1   |                                        | Edge 2
        |                                            |
        |                                            |
     Native service                              Native service
```
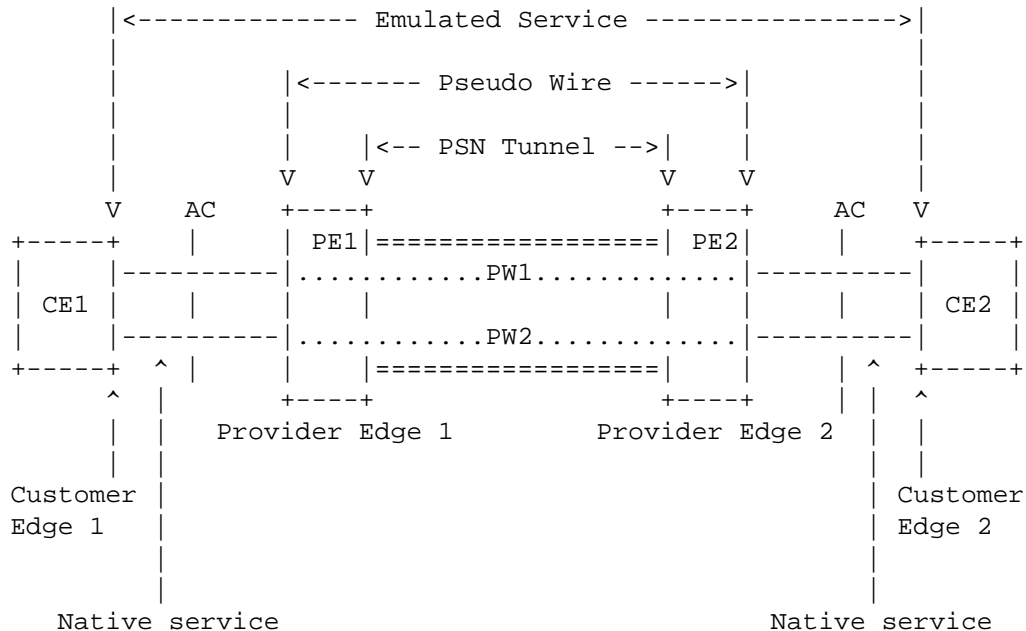
                Figure 2. PWE3 Network Reference Model

   The two PEs (PE1 and PE2) have to provide one or more PWs on behalf
   of their client CEs (CE1 and CE2) to enable the client CEs to
   communicate over the PSN. A PSN tunnel is established to provide a
   data path for the PW. The PW traffic is invisible to the core
   network, and the core network is transparent to the CEs. Native data
   units (bits, cells, or packets) arrive via the AC, are encapsulated
   in a PW-PDU, and are carried across the underlying network via the
   PSN tunnel. The PEs perform the necessary encapsulation and
   decapsulation of PW-PDUs and handle any other functions required by
   the PW service, such as sequencing or timing.

4.2. PWE3 Pre-processing

   Some applications have to perform operations on the native data units
   received from the CE (including both payload and signaling traffic)
   before they are transmitted across the PW by the PE. Examples
   include Ethernet bridging, SONET cross-connect, translation of
   locally-significant identifiers such as VCI/VPI, or translation to
   another service type. These operations could be carried out in
   external equipment, and the processed data could be sent to the PE

over one or more physical interfaces. In most cases, could be in
undertaking these operations within the PE provides cost and
operational benefits. Processed data is then presented to the PW via
a virtual interface within the PE. These pre-processing operations
are included in the PWE3 reference model to provide a common
reference point, but the detailed description of these operations is
outside the scope of the PW definition given here.

```
                         PW
                    End Service
                         |
                         |<------- Pseudo Wire ------>|
                         |                           |
                         |      |<-- PSN Tunnel -->|     |
                         V     V                   V     V     PW
                    +-----+----+             +----+ End Service
      +-----+   |PREP | PE1|=================| PE2|    |   +-----+
      |     |   |     |    |...........PW1.............|---------|   |     |
      | CE1 |----|     |    |                    |    |    | CE2 |
      |     | ^ |     |    |...........PW2.............|---------|   |     |
      +-----+ | |     |    |=================|    |    | ^ +-----+
            | +-----+----+             +----+    | |
            |       ^                             | |
            |       |                             | |
            |       |<------- Emulated Service ------->| |
            |       |                             |
            | Virtual physical                    |
            |   termination                       |
            |       ^                             |
         CE1 native |                          CE2 native
           service  |                            service
                    |
              CE2 native
                service
```
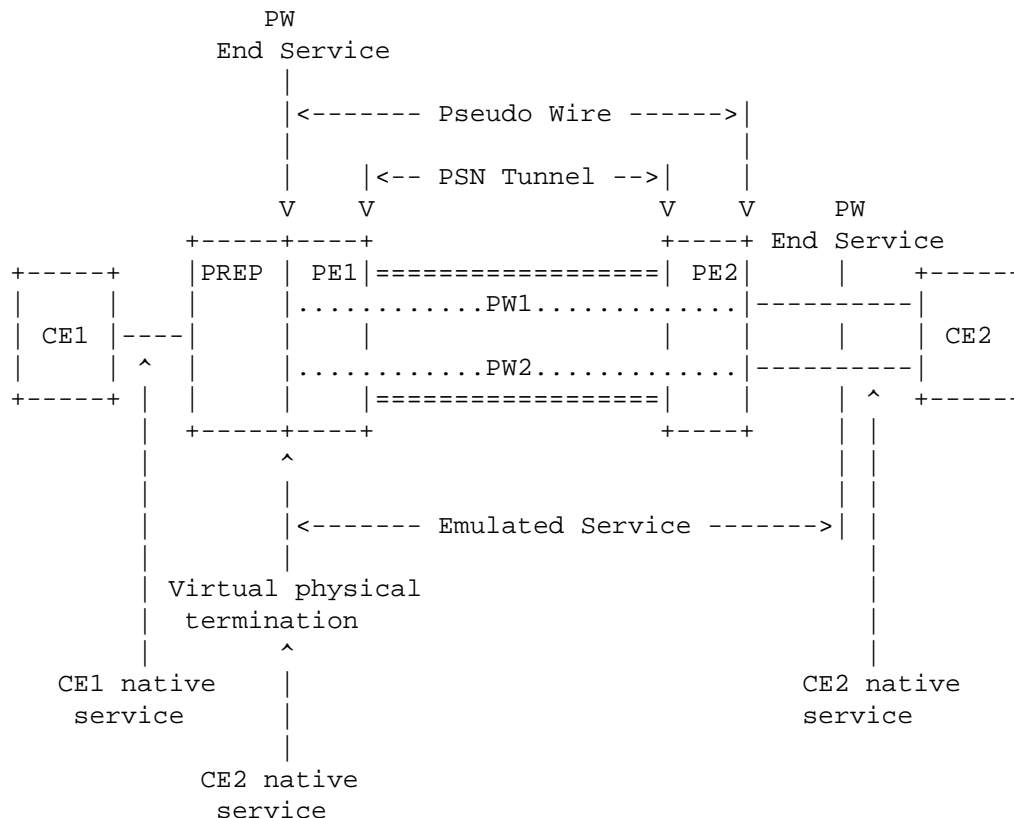
        Figure 3. Pre-processing within the PWE3 Network Reference Model

   Figure 3 shows the interworking of one PE with pre-processing (PREP),
   and a second without this functionality. This reference point
   emphasizes that the functional interface between PREP and the PW is
   that represented by a physical interface carrying the service. This
   effectively defines the necessary inter-working specification.

   The operation of a system in which both PEs include PREP
   functionality is also supported.

The required pre-processing can be divided into two components:

    o Forwarder (FWRD)
    o Native Service Processing (NSP)

4.2.1. Forwarders

Some applications have to forward payload elements selectively from
one or more ACs to one or more PWs. In such cases, there will also be
a need to perform the inverse function on PWE3-PDUs received by a PE
from the PSN. This is the function of the forwarder.

The forwarder selects the PW based on, for example, the incoming AC,
the contents of the payload, or some statically and/or dynamically
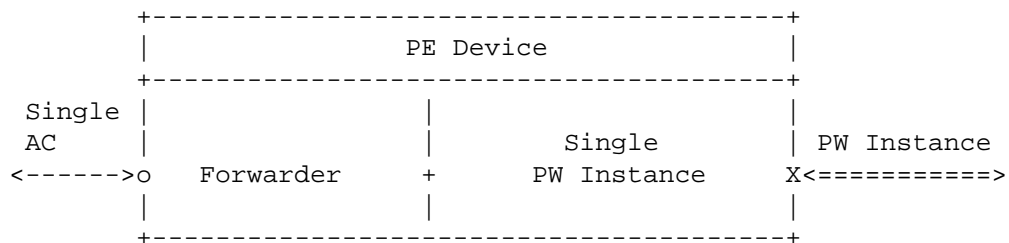configured forwarding information.

```
              +----------------------------------------+
              |               PE Device                |
              +----------------------------------------+
     Single |                  |                     |
       AC   |                  |         Single       | PW Instance
     <------>o   Forwarder     +      PW Instance     X<===========>
            |                  |                     |
              +----------------------------------------+
```

                 Figure 4a. Simple Point-to-Point Service

```
              +----------------------------------------+
              |               PE Device                |
              +----------------------------------------+
     Multiple|                  |         Single       | PW Instance
       AC    |                  +      PW Instance     X<===========>
     <------>o                  |                     |
            |                  |---------------------|
     <------>o                  |         Single       | PW Instance
            |     Forwarder     +      PW Instance     X<===========>
     <------>o                  |                     |
            |                  |---------------------|
     <------>o                  |         Single       | PW Instance
            |                  +      PW Instance     X<===========>
     <------>o                  |                     |
              +----------------------------------------+
```
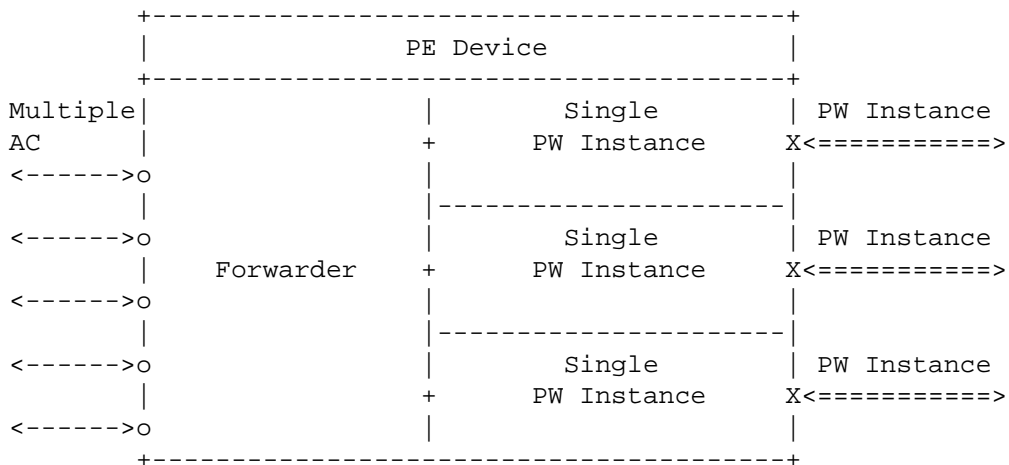
              Figure 4b. Multiple AC to Multiple PW Forwarding

Figure 4a shows a simple forwarder that performs some type of
filtering operation. Because the forwarder has a single input and a
single output interface, filtering is the only type of forwarding

operation that applies. Figure 4b shows a more general forwarding
situation where payloads are extracted from one or more ACs and
directed to one or more PWs. In this case filtering, direction, and
combination operations may be performed on the payloads. For
example, if the AC were Frame Relay, the forwarder might perform
Frame Relay switching and the PW instances might be the inter-switch
links.

4.2.2. Native Service Processing

Some applications required some form of data or address translation,
or some other operation requiring knowledge of the semantics of the
payload. This is the function of the Native Service Processor (NSP).

The use of the NSP approach simplifies the design of the PW by
restricting a PW to homogeneous operation. NSP is included in the
reference model to provide a defined interface to this functionality.
The specification of the various types of NSP is outside the scope of
PWE3.

```
              +----------------------------------------+
              |                 PE Device              |
     Multiple +----------------------------------------+
     AC       |        |         |    Single  | PW Instance
     <------>o  NSP #        +    PW Instance  X<===========>
              |        |         |            |
              |------|           |--------------------|
              |        |         |    Single  | PW Instance
     <------>o  NSP #Forwarder +  PW Instance  X<===========>
              |        |         |            |
              |------|           |--------------------|
              |        |         |    Single  | PW Instance
     <------>o  NSP #        +    PW Instance  X<===========>
              |        |         |            |
              +----------------------------------------+
```
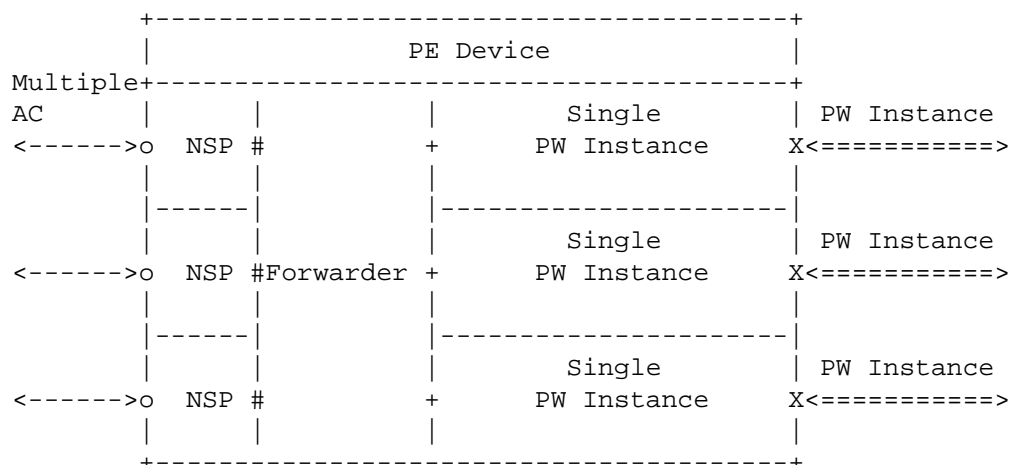
       Figure 5. NSP in a Multiple AC to Multiple PW Forwarding PE

Figure 5 illustrates the relationship between NSP, forwarder, and PWs
in a PE. The NSP function may apply any transformation operation
(modification, injection, etc.) on the payloads as they pass between
the physical interface to the CE and the virtual interface to the
forwarder. These transformation operations will, of course, be
limited to those that have been implemented in the data path, and
that are enabled by the PE configuration. A PE device may contain
more than one forwarder.

This model also supports the operation of a system in which the NSP
functionality includes terminating the data-link, and the application
of Network Layer processing to the payload.

4.3. Maintenance Reference Model

Figure 6 illustrates the maintenance reference model for PWs.

```
         |<-------- CE (end-to-end) Signaling ------>|
         |       |<---- PW/PE Maintenance ----->|    |
         |       |      |<-- PSN Tunnel -->|     |    |
         |       |      |    Signaling     |     |    |
         |       V      V  (out of scope)  V     V    |
        v      +-----+                    +-----+    v
     +-----+   | PE1 |====================| PE2 |   +-----+
     |     |---|-----|..............PW1..............|-----|   |
     | CE1 |   |     |                    |     |   | CE2 |
     |     |---|-----|..............PW2..............|-----|   |
     +-----+   |     |====================|     |   +-----+
               +-----+                    +-----+
        Customer    Provider                 Provider    Customer
         Edge 1      Edge 1                   Edge 2      Edge 2
```
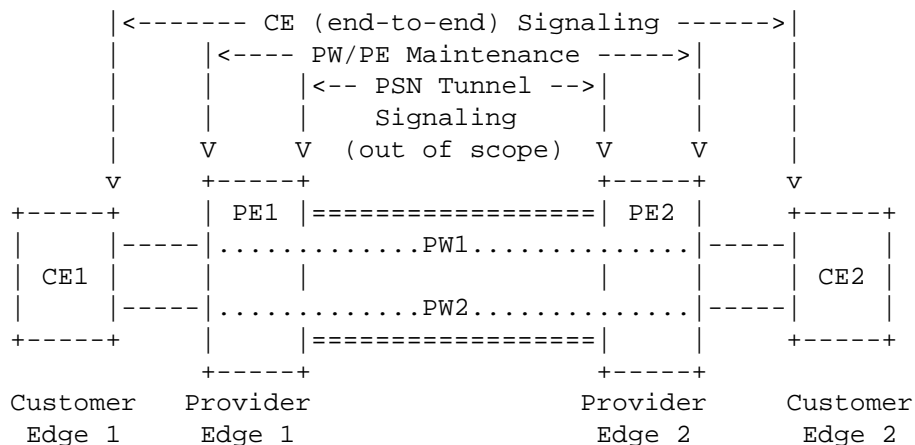
Figure 6. PWE3 Maintenance Reference Model

The following signaling mechanisms are required:

    o The CE (end-to-end) signaling is between the CEs. This
      signaling could be Frame Relay PVC status signaling, ATM SVC
      signaling, TDM CAS signaling, etc.

    o The PW/PE Maintenance is used between the PEs (or NSPs) to set
      up, maintain, and tear down PWs, including any required
      coordination of parameters.

    o The PSN Tunnel signaling controls the PW multiplexing and some
      elements of the underlying PSN. Examples are L2TP control
      protocol, MPLS LDP, and RSVP-TE. The definition of the
      information that PWE3 needs signaled is within the scope of
      PWE3, but the signaling protocol itself is not.

4.4. Protocol Stack Reference Model

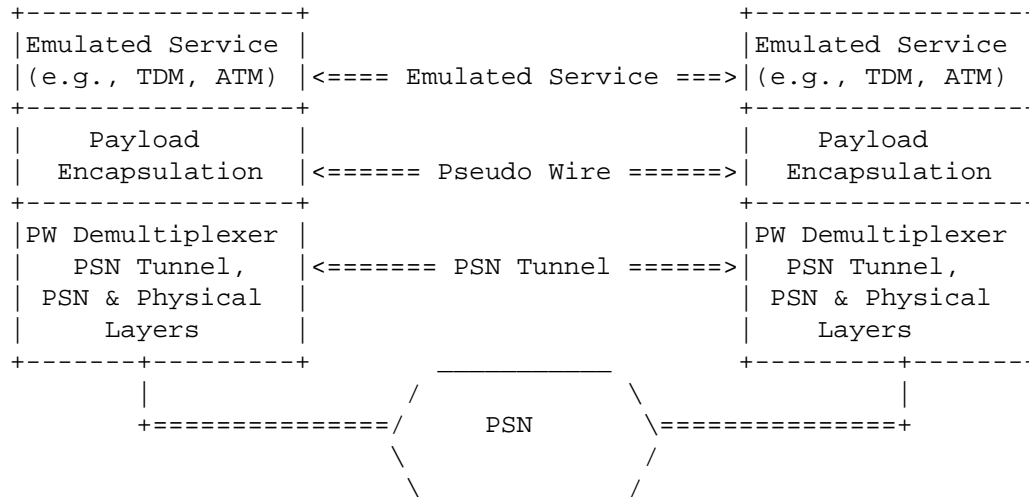   Figure 7 illustrates the protocol stack reference model for PWs.

```
   +-----------------+                                 +-----------------+
   |Emulated Service |                                 |Emulated Service |
   |(e.g., TDM, ATM) |<==== Emulated Service ===>|(e.g., TDM, ATM) |
   +-----------------+                                 +-----------------+
   |     Payload     |                                 |     Payload     |
   |  Encapsulation  |<====== Pseudo Wire ======>|  Encapsulation  |
   +-----------------+                                 +-----------------+
   |PW Demultiplexer |                                 |PW Demultiplexer |
   |    PSN Tunnel,  |<======= PSN Tunnel ======>|    PSN Tunnel,  |
   |  PSN & Physical |                                 |  PSN & Physical |
   |      Layers     |                                 |      Layers     |
   +-------+---------+             _____         +---------+-------+
           |                      /           \                |
           +==============/      PSN      \==============+
                                 \                   /
                                  _____/
```

               Figure 7. PWE3 Protocol Stack Reference Model

   The PW provides the CE with an emulated physical or virtual
   connection to its peer at the far end. Native service PDUs from the
   CE are passed through an Encapsulation Layer at the sending PE and
   then sent over the PSN. The receiving PE removes the encapsulation
   and restores the payload to its native format for transmission to the
   destination CE.

4.5. Pre-processing Extension to Protocol Stack Reference Model

   Figure 8 illustrates how the protocol stack reference model is
   extended to include the provision of pre-processing (forwarding and
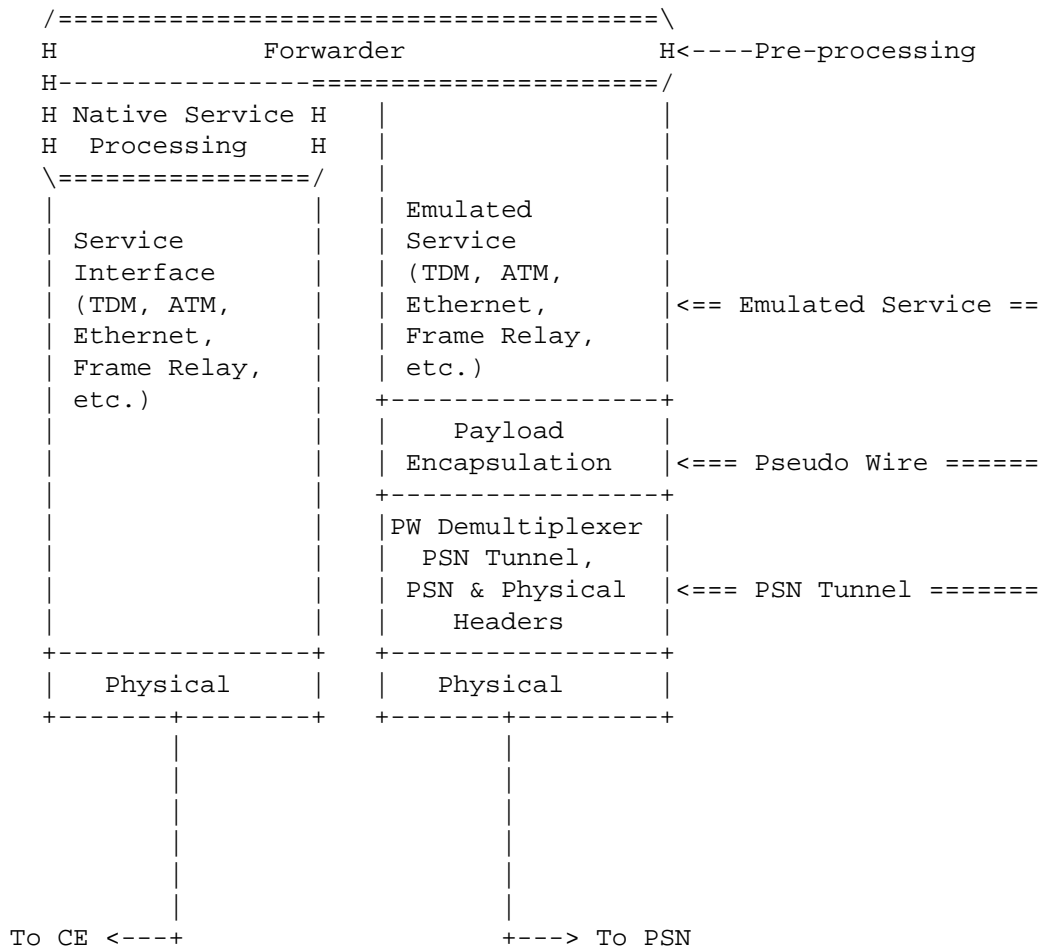   NSP). This shows the placement of the physical interface relative to
   the CE.

```
   /====================================\
   H          Forwarder               H<----Pre-processing
   H-------------======================/
   H Native Service H  |               |
   H  Processing    H  |               |
   \===============/   |               |
   |               |   | Emulated      |
   | Service       |   | Service       |
   | Interface     |   | (TDM, ATM,    |
   | (TDM, ATM,    |   | Ethernet,     |<== Emulated Service ==
   | Ethernet,     |   | Frame Relay,  |
   | Frame Relay,  |   | etc.)         |
   | etc.)         |   +---------------+
   |               |   |    Payload     |
   |               |   | Encapsulation |<=== Pseudo Wire ======
   |               |   +---------------+
   |               |   |PW Demultiplexer |
   |               |   |  PSN Tunnel,   |
   |               |   | PSN & Physical |<=== PSN Tunnel =======
   |               |   |    Headers     |
   +---------------+   +---------------+
   |  Physical     |   |  Physical     |
   +-------+-------+   +-------+---------+
           |                   |
           |                   |
           |                   |
           |                   |
           |                   |
           |                   |
   To CE <---+                 +---> To PSN
```

       Figure 8. Protocol Stack Reference Model with Pre-processing

5. PW Encapsulation

   The PW Encapsulation Layer provides the necessary infrastructure to
   adapt the specific payload type being transported over the PW to the
   PW Demultiplexer Layer used to carry the PW over the PSN.

   The PW Encapsulation Layer consists of three sub-layers:

        o Payload Convergence
        o Timing
        o Sequencing

   The PW Encapsulation sub-layering and its context with the protocol
   stack are shown in Figure 9.

```
       +--------------------------+
       |          Payload         |
       /==========================\ <------ Encapsulation
       H    Payload Convergence   H         Layer
       H--------------------------H
       H           Timing         H
       H--------------------------H
       H         Sequencing       H
       \==========================/
       |     PW Demultiplexer     |
       +--------------------------+
       |     PSN Convergence      |
       +--------------------------+
       |           PSN            |
       +--------------------------+
       |         Data-Link        |
       +--------------------------+
       |         Physical         |
       +--------------------------+
```
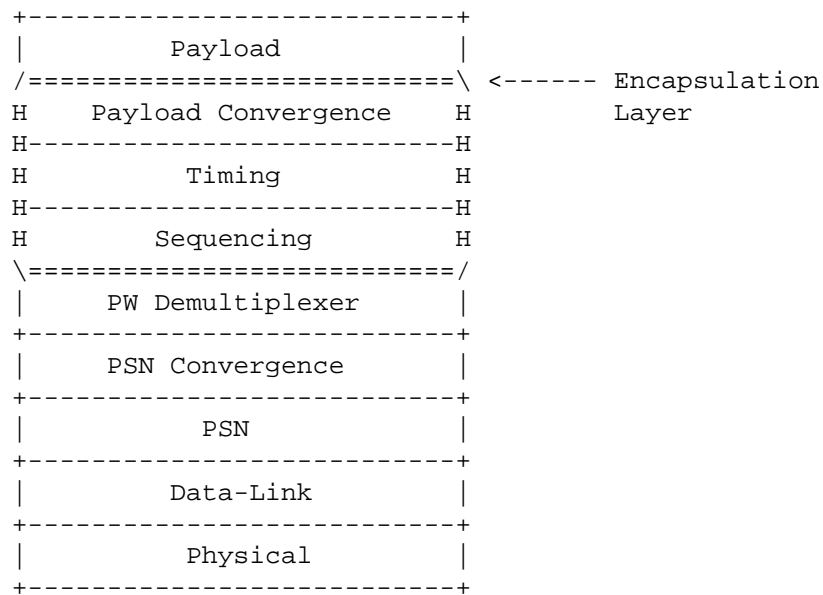
                Figure 9. PWE3 Encapsulation Layer in Context

   The Payload Convergence sub-layer is highly tailored to the specific
   payload type. However grouping a number of target payload types into
   a generic class, and then providing a single convergence sub-layer
   type common to the group, reduces the number of payload convergence
   sub-layer types. This decreases implementation complexity. The
   provision of per-packet signaling and other out-of-band information
   (other than sequencing or timing) is undertaken by this layer.

   The Timing and Sequencing Layers provide generic services to the
   Payload Convergence Layer for all payload types that require them.

5.1. Payload Convergence Layer

5.1.1. Encapsulation

   The primary task of the Payload Convergence Layer is the
   encapsulation of the payload in PW-PDUs. The native data units to be
   encapsulated may contain an L2 header or L1 overhead. This is
   service specific. The Payload Convergence header carries the
   additional information needed to replay the native data units at the
   CE-bound physical interface. The PW Demultiplexer header is not
   considered part of the PW header.

Not all the additional information needed to replay the native data
units have to be carried in the PW header of the PW PDUs. Some
information (e.g., service type of a PW) may be stored as state
information at the destination PE during PW set up.

5.1.2. PWE3 Channel Types

The PW Encapsulation Layer and its associated signaling require one
or more of the following types of channels from its underlying PW
Demultiplexer and PSN Layers (channel type 1 plus one or more of
channel types 2 through 4):

1. A reliable control channel for signaling line events, status
   indications, and, in exceptional cases, CE-CE events that must be
   translated and sent reliably between PEs. PWE3 may need this type
   of control channel to provide faithful emulation of complex data-
   link protocols.

2. A high-priority, unreliable, sequenced channel. A typical use is
   for CE-to-CE signaling. "High priority" may simply be indicated
   via the DSCP bits for IP or the EXP bits for MPLS, giving the
   packet priority during transit. This channel type could also use
   a bit in the tunnel header itself to indicate that packets
   received at the PE should be processed with higher priority
   [RFC2474].

3. A sequenced channel for data traffic that is sensitive to packet
   reordering (one classification for use could be for any non-IP
   traffic).

4. An unsequenced channel for data traffic insensitive to packet
   order.

The data channels (2, 3, and 4 above) should be carried "in band"
with one another to as much of a degree as is reasonably possible on
a PSN.

Where end-to-end connectivity may be disrupted by address translation
[RFC3022], access-control lists, firewalls, etc., the control channel
may be able to pass traffic and setup the PW, while the PW data
traffic is blocked by one or more of these mechanisms. In these
cases unless the control channel is also carried "in band", the
signaling to set up the PW will not confirm the existence of an end-
to-end data path. In some cases there is a need to synchronize CE
events with the data carried over a PW. This is especially the case

with TDM circuits (e.g., the on-hook/off-hook events in PSTN switches
might be carried over a reliable control channel whereas the
associated bit stream is carried over a sequenced data channel).

PWE3 channel types that are not needed by the supported PWs need not
be included in such an implementation.

5.1.3. Quality of Service Considerations

Where possible, it is desirable to employ mechanisms to provide PW
Quality of Service (QoS) support over PSNs.

5.2. Payload-Independent PW Encapsulation Layers

Two PWE3 Encapsulation sub-layers provide common services to all
payload types: Sequencing and Timing. These services are optional
and are only used if a particular PW instance needs them. If the
service is not needed, the associated header may be omitted in order
to conserve processing and network resources.

Sometimes a specific payload type will require transport with or
without sequence and/or real-time support. For example, an invariant
of Frame Relay transport is the preservation of packet order. Some
Frame Relay applications expect delivery in order and may not cope
with reordering of the frames. However, where the Frame Relay
service is itself only being used to carry IP, it may be desirable to
relax this constraint to reduce per-packet processing cost.

The guiding principle is that, when possible, an existing IETF
protocol should be used to provide these services. When a suitable
protocol is not available, the existing protocol should be extended
or modified to meet the PWE3 requirements, thereby making that
protocol available for other IETF uses. In the particular case of
timing, more than one general method may be necessary to provide for
the full scope of payload timing requirements.

5.2.1. Sequencing

The sequencing function provides three services: frame ordering,
frame duplication detection, and frame loss detection. These
services allow the emulation of the invariant properties of a
physical wire. Support for sequencing depends on the payload type
and may be omitted if it is not needed.

The size of the sequence-number space depends on the speed of the
emulated service, and on the maximum time of the transient conditions
in the PSN. A sequence number space greater than 2^16 may therefore
be needed to prevent the sequence number space from wrapping during
the transient.

5.2.1.1. Frame Ordering

When packets carrying the PW-PDUs traverse a PSN, they may arrive out
of order at the destination PE. For some services, the frames
(control frames, data frames, or both) must be delivered in order.
For these services, some mechanism must be provided for ensuring in-
order delivery. Providing a sequence number in the sequence sub-
layer header for each packet is one possible approach.
Alternatively, it can be noted that sequencing is a subset of the
problem of delivering timed packets, and that a single combined
mechanism such as [RFC3550] may be employed.

There are two possible misordering strategies:

    o Drop misordered PW PDUs.

    o Try to sort PW PDUs into the correct order.

The choice of strategy will depend on

    o how critical the loss of packets is to the operation of the PW
      (e.g., the acceptable bit error rate),

    o the speeds of the PW and PSN,

    o the acceptable delay (as delay must be introduced to reorder),
      and

    o the expected incidence of misordering.

5.2.1.2. Frame Duplication Detection

In rare cases, packets traversing a PW may be duplicated by the
underlying PSN. For some services, frame duplication is not
acceptable. For these services, some mechanism must be provided to
ensure that duplicated frames will not be delivered to the
destination CE. The mechanism may be the same as that used to ensure
in-order frame delivery.

5.2.1.3. Frame Loss Detection

   A destination PE can determine whether a frame has been lost by
   tracking the sequence numbers of the PW PDUs received.

   In some instances, if a PW PDU fails to arrive within a certain time,
   a destination PE will have to presume that it is lost. If a PW-PDU
   that has been processed as lost subsequently arrives, the destination
   PE must discard it.

5.2.2. Timing

   A number of native services have timing expectations based on the
   characteristics of the networks they were designed to travel over.
   The emulated service may have to duplicate these network
   characteristics as closely as possible: e.g., in delivering native
   traffic with bitrate, jitter, wander, and delay characteristics
   similar to those received at the sending PE.

   In such cases, the receiving PE has to play out the native traffic as
   it was received at the sending PE. This relies on timing information
   either sent between the two PEs, or in some cases received from an
   external reference.

   Therefore, Timing Sub-layer must support two timing functions: clock
   recovery and timed payload delivery. A particular payload type may
   require either or both of these services.

5.2.2.1. Clock Recovery

   Clock recovery is the extraction of output transmission bit timing
   information from the delivered packet stream, and it requires a
   suitable mechanism. A physical wire carries the timing information
   natively, but extracting timing from a highly jittered source, such
   as packet stream, is a relatively complex task. Therefore, it is
   desirable that an existing real-time protocol such as [RFC3550] be
   used for this purpose, unless it can be shown that this is unsuitable
   or unnecessary for a particular payload type.

5.2.2.2. Timed Delivery

   Timed delivery is the delivery of non-contiguous PW PDUs to the PW
   output interface with a constant phase relative to the input
   interface. The timing of the delivery may be relative to a clock
   derived from the packet stream received over the PSN clock recovery,
   or to an external clock.

5.3. Fragmentation

   Ideally, a payload would be relayed across the PW as a single unit.
   However, there will be cases where the combined size of the payload
   and its associated PWE3 and PSN headers will exceed the PSN path MTU.
   When a packet size exceeds the MTU of a given network, fragmentation
   and reassembly have to be performed for the packet to be delivered.
   Since fragmentation and reassembly generally consume considerable
   network resources, as compared to simply switching a packet in its
   entirety, the need for fragmentation and reassembly throughout a
   network should be reduced or eliminated to the extent possible. Of
   particular concern for fragmentation and reassembly are aggregation
   points where large numbers of PWs are processed (e.g., at the PE).

   Ideally, the equipment originating the traffic sent over the PW will
   have adaptive measures in place (e.g., [RFC1191], [RFC1981]) that
   ensure that packets needing to be fragmented are not sent. When this
   fails, the point closest to the sending host with fragmentation and
   reassembly capabilities should attempt to reduce the size of packets
   to satisfy the PSN MTU. Thus, in the reference model for PWE3
   (Figure 3), fragmentation should first be performed at the CE if
   possible. Only if the CE cannot adhere to an acceptable MTU size for
   the PW should the PE attempt its own fragmentation method.

   In cases where MTU management fails to limit the payload to a size
   suitable for transmission of the PW, the PE may fall back to either a
   generic PW fragmentation method or, if available, the fragmentation
   service of the underlying PSN.

   It is acceptable for a PE implementation not to support
   fragmentation. A PE that does not will drop packets that exceed the
   PSN MTU, and the management plane of the encapsulating PE may be
   notified.

   If the length of a L2/L1 frame, restored from a PW PDU, exceeds the
   MTU of the destination AC, it must be dropped. In this case, the
   management plane of the destination PE may be notified.

5.4. Instantiation of the Protocol Layers

   This document does not address the detailed mapping of the Protocol
   Layering model to existing or future IETF standards. The
   instantiation of the logical Protocol Layering model is shown in
   Figure 9.

5.4.1. PWE3 over an IP PSN

   The protocol definition of PWE3 over an IP PSN should employ existing
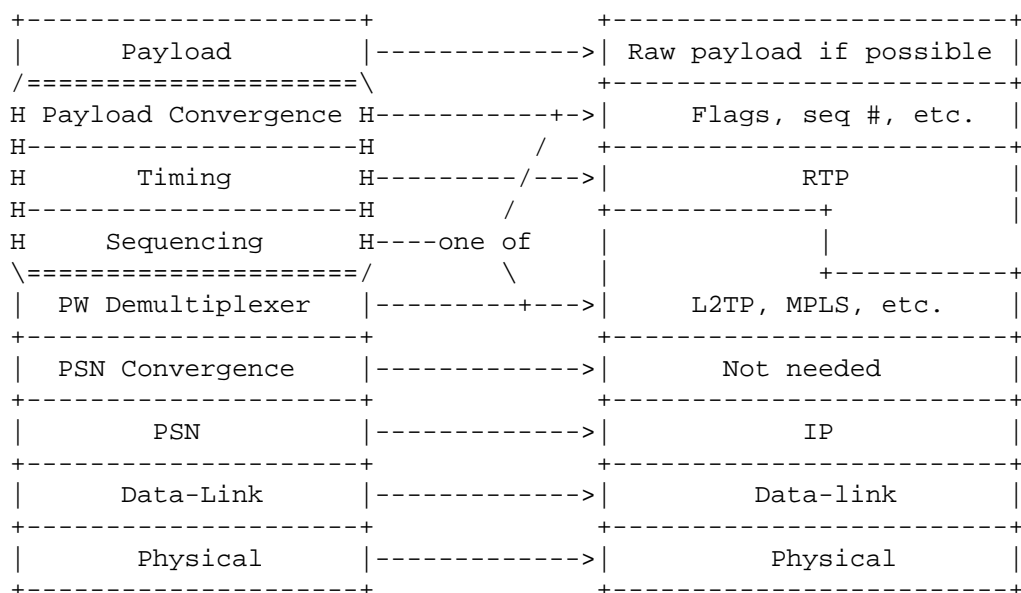   IETF protocols where possible.

```
   +--------------------+                +------------------------+
   |     Payload        |------------->| Raw payload if possible |
   /====================\               +------------------------+
   H Payload Convergence H-----------+->|    Flags, seq #, etc.  |
   H--------------------H        /    +------------------------+
   H      Timing        H--------/--->|          RTP           |
   H--------------------H      /    +-------------+          |
   H     Sequencing     H----one of  |           |          |
   \====================/       \    |        +-----------+
   |   PW Demultiplexer  |---------+--->|    L2TP, MPLS, etc.    |
   +--------------------+                +------------------------+
   |   PSN Convergence   |------------->|       Not needed       |
   +--------------------+                +------------------------+
   |        PSN         |------------->|           IP           |
   +--------------------+                +------------------------+
   |     Data-Link      |------------->|       Data-link        |
   +--------------------+                +------------------------+
   |     Physical       |------------->|        Physical        |
   +--------------------+                +------------------------+
```

                    Figure 10. PWE3 over an IP PSN

   Figure 10 shows the protocol layering for PWE3 over an IP PSN. As a
   rule, the payload should be carried as received from the NSP, with
   the Payload Convergence Layer provided when needed. However, in
   certain circumstances it may be justifiable to transmit the payload
   in some processed form. The reasons for this must be documented in
   the Encapsulation Layer definition for that payload type.

   Where appropriate, explicit timing is provided by RTP [RFC3550],
   which, when used, also provides a sequencing service. When the PSN
   is UDP/IP, the RTP header follows the UDP header and precedes the PW
   control field. For all other cases the RTP header follows the PW
   control header.

   The encapsulation layer may additionally carry a sequence number.
   Sequencing is to be provided either by RTP or by the PW encapsulation
   layer, but not by both.

PW Demultiplexing is provided by the PW label, which may take the
form specified in a number of IETF protocols; e.g., an MPLS label
[MPLSIP], an L2TP session ID [RFC3931], or a UDP port number
[RFC768]. When PWs are carried over IP, the PSN Convergence Layer
will not be needed.

As a special case, if the PW Demultiplexer is an MPLS label, the
protocol architecture of section 5.4.2 can be used instead of the
protocol architecture of this section.

5.4.2. PWE3 over an MPLS PSN

The MPLS ethos places importance on wire efficiency. By using a
control word, some components of the PWE3 protocol layers can be
compressed to increase this efficiency.

```
+--------------------+
|      Payload       |
/====================\
H Payload Convergence H--+
H--------------------H  |       +--------------------------------+
H      Timing        H--------->|              RTP               |
H--------------------H  |       +--------------------------------+
H     Sequencing     H--+------>| Flags, Frag, Len, Seq #, etc   |
\====================/  |       +--------------------------------+
|   PW Demultiplexer  |--------->|             PW Label           |
+--------------------+  |       +--------------------------------+
|  PSN Convergence    |--+ +--->| Outer Label or MPLS-in-IP encap|
+--------------------+    |     +--------------------------------+
|        PSN          |-----+
+--------------------+
|      Data-Link      |
+--------------------+
|      Physical       |
+--------------------+
```
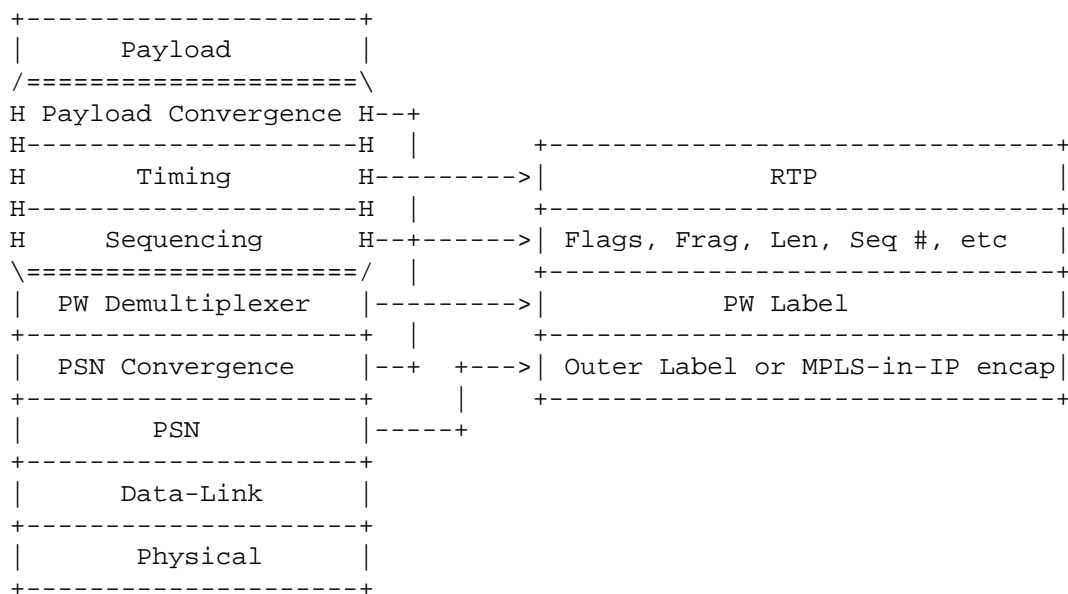
          Figure 11. PWE3 over an MPLS PSN Using a Control Word

Figure 11 shows the protocol layering for PWE3 over an MPLS PSN. An
inner MPLS label is used to provide the PW demultiplexing function.
A control word is used to carry most of the information needed by the
PWE3 Encapsulation Layer and the PSN Convergence Layer in a compact
format. The flags in the control word provide the necessary payload
convergence. A sequence field provides support for both in-order
payload delivery and a PSN fragmentation service within the PSN
Convergence Layer (supported by a fragmentation control method).
Ethernet pads all frames to a minimum size of 64 bytes. The MPLS
header does not include a length indicator. Therefore, to allow PWE3

to be carried in MPLS to pass correctly over an Ethernet data-link, a
length correction field is needed in the control word. As with an IP
PSN, where appropriate, timing is provided by RTP [RFC3550].

In some networks, it may be necessary to carry PWE3 over MPLS over
IP. In these circumstances, the PW is encapsulated for carriage over
MPLS as described in this section, and then a method of carrying MPLS
over an IP PSN (such as GRE [RFC2784], [RFC2890]) is applied to the
resultant PW-PDU.

5.4.3. PW-IP Packet Discrimination

For MPLS PSNs, there is an additional constraint on the PW packet
format. Some label switched routers detect IP packets based on the
initial four bits of the packet content. To facilitate proper
functioning, these bits in PW packets must not be the same as an IP
version number in current use.

6. PW Demultiplexer Layer and PSN Requirements

PWE3 places three service requirements on the protocol layers used to
carry it across the PSN:

     o Multiplexing
     o Fragmentation
     o Length and Delivery

6.1. Multiplexing

The purpose of the PW Demultiplexer Layer is to allow multiple PWs to
be carried in a single tunnel. This minimizes complexity and
conserves resources.

Some types of native service are capable of grouping multiple
circuits into a "trunk"; e.g., multiple ATM VCs in a VP, multiple
Ethernet VLANs on a physical media, or multiple DS0 services within a
T1 or E1. A PW may interconnect two end-trunks. That trunk would
have a single multiplexing identifier.

When a MPLS label is used as a PW Demultiplexer, setting of the TTL
value [RFC3032] in the PW label is application specific.

6.2. Fragmentation

   If the PSN provides a fragmentation and reassembly service of
   adequate performance, it may be used to obtain an effective MTU that
   is large enough to transport the PW PDUs. See section 5.3 for a full
   discussion of the PW fragmentation issues.

6.3. Length and Delivery

   PDU delivery to the egress PE is the function of the PSN Layer.

   If the underlying PSN does not provide all the information necessary
   to determine the length of a PW-PDU, the Encapsulation Layer must
   provide it.

6.4. PW-PDU Validation

   It is a common practice to use an error detection mechanism such as a
   CRC or similar mechanism to ensure end-to-end integrity of frames.
   The PW service-specific mechanisms must define whether the packet's
   checksum shall be preserved across the PW or be removed from PE-bound
   PDUs and then be recalculated for insertion in CE-bound data.

   The former approach saves work, whereas the latter saves bandwidth.
   For a given implementation, the choice may be dictated by hardware
   restrictions, which may not allow the preservation of the checksum.

   For protocols such as ATM and FR, the scope of the checksum is
   restricted to a single link. This is because the circuit identifiers
   (e.g., FR DLCI or ATM VPI/VCI) only have local significance and are
   changed on each hop or span. If the circuit identifier (and thus
   checksum) were going to change as part of the PW emulation, it would
   be more efficient to strip and recalculate the checksum.

   The service-specific document for each protocol must describe the
   validation scheme to be used.

6.5. Congestion Considerations

   The PSN carrying the PW may be subject to congestion. The congestion
   characteristics will vary with the PSN type, the network architecture
   and configuration, and the loading of the PSN.

   If the traffic carried over the PW is known to be TCP friendly (by,
   for example, packet inspection), packet discard in the PSN will
   trigger the necessary reduction in offered load, and no additional
   congestion avoidance action is necessary.

If the PW is operating over a PSN that provides enhanced delivery,
the PEs should monitor packet loss to ensure that the requested
service is actually being delivered. If it is not, then the PE
should assume that the PSN is providing a best-effort service and
should use the best-effort service congestion avoidance measures
described below.

If best-effort service is being used and the traffic is not known to
be TCP friendly, the PEs should monitor packet loss to ensure that
the loss rate is within acceptable parameters. Packet loss is
considered acceptable if a TCP flow across the same network path and
experiencing the same network conditions would achieve an average
throughput, measured on a reasonable timescale, not less than that
which the PW flow is achieving. This condition can be satisfied by
implementing a rate-limiting measure in the NSP, or by shutting down
one or more PWs. The choice of which approach to use depends upon
the type of traffic being carried. Where congestion is avoided by
shutting down a PW, a suitable mechanism must be provided to prevent
it from immediately returning to service and causing a series of
congestion pulses.

The comparison to TCP cannot be specified exactly but is intended as
an "order-of-magnitude" comparison in timescale and throughput. The
timescale on which TCP throughput is measured is the round-trip time
of the connection. In essence, this requirement states that it is
not acceptable to deploy an application (using PWE3 or any other
transport protocol) on the best-effort Internet, which consumes
bandwidth arbitrarily and does not compete fairly with TCP within an
order of magnitude. One method of determining an acceptable PW
bandwidth is described in [RFC3448].

7. Control Plane

   This section describes PWE3 control plane services.

7.1. Setup or Teardown of Pseudo Wires

   A PW must be set up before an emulated service can be established and
   must be torn down when an emulated service is no longer needed.

   Setup or teardown of a PW can be triggered by an operator command,
   from the management plane of a PE, by signaling set-up or teardown of
   an AC (e.g., an ATM SVC), or by an auto-discovery mechanism.

   During the setup process, the PEs have to exchange information (e.g.,
   learn each other's capabilities). The tunnel signaling protocol may
   be extended to provide mechanisms that enable the PEs to exchange all
   necessary information on behalf of the PW.

Manual configuration of PWs can be considered a special kind of
signaling and is allowed.

7.2. Status Monitoring

Some native services have mechanisms for status monitoring. For
example, ATM supports OAM for this purpose. For these services, the
corresponding emulated services must specify how to perform status
monitoring.

7.3. Notification of Pseudo Wire Status Changes

7.3.1. Pseudo Wire Up/Down Notification

If a native service requires bi-directional connectivity, the
corresponding emulated service can only be signaled as being up when
the PW and PSN tunnels (if used), are functional in both directions.

Because the two CEs of an emulated service are not adjacent, a
failure may occur at a place so that one or both physical links
between the CEs and PEs remain up. For example, in Figure 2, if the
physical link between CE1 and PE1 fails, the physical link between
CE2 and PE2 will not be affected and will remain up. Unless CE2 is
notified about the remote failure, it will continue to send traffic
over the emulated service to CE1. Such traffic will be discarded at
PE1. Some native services have failure notification so that when the
services fail, both CEs will be notified. For these native services,
the corresponding PWE3 service must provide a failure notification
mechanism.

Similarly, if a native service has notification mechanisms so that
all the affected services will change status from "Down" to "Up" when
a network failure is fixed, the corresponding emulated service must
provide a similar mechanism for doing so.

These mechanisms may already be built into the tunneling protocol.
For example, the L2TP control protocol [RFC2661] [RFC3931] has this
capability, and LDP has the ability to withdraw the corresponding
MPLS label.

7.3.2. Misconnection and Payload Type Mismatch

With PWE3, misconnection and payload type mismatch can occur.
Misconnection can breach the integrity of the system. Payload
mismatch can disrupt the customer network. In both instances, there
are security and operational concerns.

   The services of the underlying tunneling mechanism and its associated
   control protocol can be used to mitigate this. As part of the PW
   setup, a PW-TYPE identifier is exchanged. This is then used by the
   forwarder and the NSP to verify the compatibility of the ACs.

7.3.3. Packet Loss, Corruption, and Out-of-Order Delivery

   A PW can incur packet loss, corruption, and out-of-order delivery on
   the PSN path between the PEs. This can affect the working condition
   of an emulated service. For some payload types, packet loss,
   corruption, and out-of-order delivery can be mapped either to a bit
   error burst, or to loss of carrier on the PW. If a native service
   has some mechanism to deal with bit error, the corresponding PWE3
   service should provide a similar mechanism.

7.3.4. Other Status Notification

   A PWE3 approach may provide a mechanism for other status
   notifications, if any are needed.

7.3.5. Collective Status Notification

   The status of a group of emulated services may be affected
   identically by a single network incident. For example, when the
   physical link (or sub-network) between a CE and a PE fails, all the
   emulated services that go through that link (or sub-network) will
   fail. It is likely that a group of emulated services all terminate
   at a remote CE. There may also be multiple such CEs affected by the
   failure. Therefore, it is desirable that a single notification
   message be used to notify failure of the whole group of emulated
   services.

   A PWE3 approach may provide a mechanism for notifying status changes
   of a group of emulated circuits. One possible method is to associate
   each emulated service with a group ID when the PW for that emulated
   service is set up. Multiple emulated services can then be grouped by
   associating them with the same group ID. In status notification,
   this group ID can be used to refer all the emulated services in that
   group. The group ID mechanism should be a mechanism provided by the
   underlying tunnel signaling protocol.

7.4. Keep-Alive

   If a native service has a keep-alive mechanism, the corresponding
   emulated service must provide a mechanism to propagate it across the
   PW. Transparently transporting keep-alive messages over the PW would
   follow the principle of minimum intervention. However, to reproduce

   the semantics of the native mechanism accurately, some PWs may
   require an alternative approach, such as piggy-backing on the PW
   signaling mechanism.

7.5. Handling Control Messages of the Native Services

   Some native services use control messages for circuit maintenance.
   These control messages may be in-band (e.g., Ethernet flow control,
   ATM performance management, or TDM tone signaling) or out-of-band,
   (e.g., the signaling VC of an ATM VP, or TDM CCS signaling).

   Given the principle of minimum intervention, it is desirable that the
   PEs participate as little as possible in the signaling and
   maintenance of the native services. This principle should not,
   however, override the need to emulate the native service
   satisfactorily.

   If control messages are passed through, it may be desirable to send
   them by using either a higher priority or a reliable channel provided
   by the PW Demultiplexer layer. See Section 5.1.2, PWE3 Channel
   Types.

8. Management and Monitoring

   This section describes the management and monitoring architecture for
   PWE3.

8.1. Status and Statistics

   The PE should report the status of the interface and tabulate
   statistics that help monitor the state of the network and help
   measure service-level agreements (SLAs). Typical counters include
   the following:

      o Counts of PW-PDUs sent and received, with and without errors.
      o Counts of sequenced PW-PDUs lost.
      o Counts of service PDUs sent and received over the PSN, with and
        without errors (non-TDM).
      o Service-specific interface counts.
      o One-way delay and delay variation.

   These counters would be contained in a PW-specific MIB, and they
   should not replicate existing MIB counters.

8.2. PW SNMP MIB Architecture

   This section describes the general architecture for SNMP MIBs used to
   manage PW services and the underlying PSN. The intent here is to
   provide a clear picture of how all the pertinent MIBs fit together to
   form a cohesive management framework for deploying PWE3 services.
   Note that the names of MIB modules used below are suggestions and do
   not necessarily require that the actual modules used to realize the
   components in the architecture be named exactly so.

8.2.1. MIB Layering

   The SNMP MIBs created for PWE3 should fit the architecture shown in
   Figure 12. The architecture provides a layered modular model into
   which any supported emulated service can be connected to any
   supported PSN type. This model fosters reuse of as much
   functionality as possible. For instance, the emulated service layer
   MIB modules do not redefine the existing emulated service MIB module;
   rather, they only associate it with the pseudo wires used to carry
   the emulated service over the configured PSN. In this way, the PWE3
   MIB architecture follows the overall PWE3 architecture.

   The architecture does allow for the joining of unsupported emulated
   service or PSN types by simply defining additional MIB modules to
   associate new types with existing ones. These new modules can
   subsequently be standardized. Note that there is a separate MIB
   module for each emulated service, as well as one for each underlying
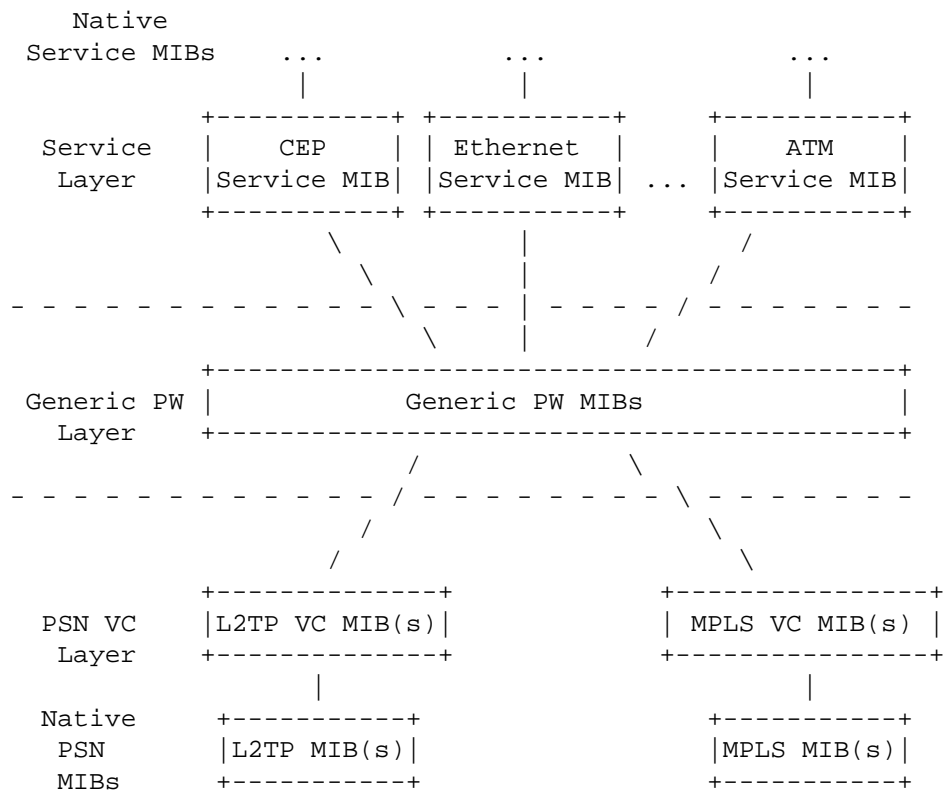   PSN. These MIB modules may be used in various combinations as
   needed.

```
      Native
   Service MIBs      ...              ...                    ...
                      |                |                      |
                 +-----------+ +-----------+         +-----------+
    Service      |    CEP    | | Ethernet  |         |    ATM    |
    Layer        |Service MIB| |Service MIB|   ...   |Service MIB|
                 +-----------+ +-----------+         +-----------+
                       \           |             /
                        \          |            /
   - - - - - - - - - - - - \ - - - | - - - - / - - - - - - -
                            \      |        /
                 +---------------------------------------------+
   Generic PW |  |             Generic PW MIBs                 |
    Layer        +---------------------------------------------+
                              /                \
   - - - - - - - - - - - - / - - - - - - - \ - - - - - - -
                        /                     \
                      /                         \
                 +--------------+          +---------------+
    PSN VC       |L2TP VC MIB(s)|          | MPLS VC MIB(s) |
    Layer        +-------------+           +---------------+
                       |                          |
    Native      +-----------+           +-----------+
     PSN        |L2TP MIB(s)|           |MPLS MIB(s)|
    MIBs        +-----------+           +-----------+
```

Figure 12. MIB Module Layering Relationship

Figure 13 shows an example for a SONET PW carried over MPLS Traffic
Engineering Tunnel and an LDP-signaled LSP.

```
                         +-----------------+
                         |    SONET MIB    |  RFC3592
                         +-----------------+
                                 |
                 +-------------------------------+
   Service       | Circuit Emulation Service MIB|
    Layer        +-----------------------------+
 - - - - - - - - - - - - | - - - - - - - - - - - - -
                         +-----------------+
   Generic PW            | Generic PW MIB  |
     Layer               +-----------------+
 - - - - - - - - - - - - | - - - - - - - - - - - - -
                         +-----------------+
    PSN VC               |   MPLS VC MIBs  |
    Layer                +-----------------+
                             |         |
             +-----------------+  +------------------+
             | MPLS-TE-STD-MIB |  | MPLS-LSR-STD-MIB |
             +-----------------+  +------------------+
```
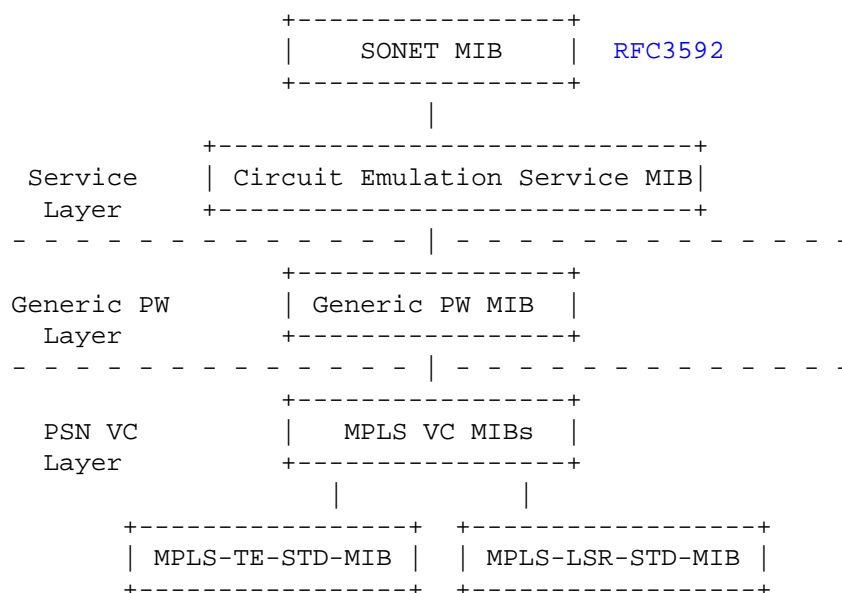
         Figure 13. SONET PW over MPLS PSN Service-Specific Example

8.2.2. Service Layer MIB Modules

   This conceptual layer in the model contains MIB modules used to
   represent the relationship between emulated PWE3 services such as
   Ethernet, ATM, or Frame Relay and the pseudo-wire used to carry that
   service across the PSN. This layer contains corresponding MIB
   modules used to mate or adapt those emulated services to the generic
   pseudo-wire representation these are represented in the "Generic PW
   MIB" functional block in Figure 13 above. This working group should
   not produce any MIB modules for managing the general service; rather,
   it should produce just those modules used to interface or adapt the
   emulated service onto the PWE3 management framework as shown above.
   For example, the standard SONET-MIB [RFC3592] is designed and
   maintained by another working group. The SONET-MIB is designed to
   manage the native service without PW emulation. However, the PWE3
   working group is chartered to produce standards that show how to
   emulate existing technologies such as SONET/SDH over pseudo-wires
   rather than reinvent those modules.

8.2.3. Generic PW MIB Modules

   The middle layer in the architecture is referred to as the Generic PW
   Layer. MIBs in this layer are responsible for providing pseudo-wire
   specific counters and service models used for monitoring and
   configuration of PWE3 services over any supported PSN service. That
   is, this layer provides a general model of PWE3 abstraction for
   management purposes. This MIB is used to interconnect the MIB
   modules residing in the Service Layer to the PSN VC Layer MIBs (see
   section 8.2.4).

8.2.4. PSN VC Layer MIB Modules

   The third layer in the PWE3 management architecture is referred to as
   the PSN VC Layer. It is composed of MIBs that are specifically
   designed to associate pseudo-wires onto those underlying PSN
   transport technologies that carry the pseudo-wire payloads across the
   PSN. In general, this means that the MIB module provides a mapping
   between the emulated service that is mapped to the pseudo-wire via
   the Service Layer and the Generic PW MIB Layer onto the native PSN
   service. For example, in the case of MPLS, for example, it is
   required that the general VC service be mapped into MPLS LSPs via the
   MPLS-LSR-STD-MIB [RFC3813] or Traffic-Engineered (TE) Tunnels via the
   MPLS-TE-STD-MIB [RFC3812]. In addition, the MPLS-LDP-STD-MIB
   [RFC3815] may be used to reveal the MPLS labels that are distributed
   over the MPLS PSN in order to maintain the PW service. As with the
   native service MIB modules described earlier, the MIB modules used to
   manage the native PSN services are produced by other working groups
   that design and specify the native PSN services. These MIBs should
   contain the appropriate mechanisms for monitoring and configuring the
   PSN service that the emulated PWE3 service will function correctly.

8.3. Connection Verification and Traceroute

   A connection verification mechanism should be supported by PWs.
   Connection verification and other alarm mechanisms can alert the
   operator that a PW has lost its remote connection. The opaque nature
   of a PW means that it is not possible to specify a generic connection
   verification or traceroute mechanism that passes this status to the
   CEs over the PW. If connection verification status of the PW is
   needed by the CE, it must be mapped to the native connection status
   method.

   For troubleshooting purposes, it is sometimes desirable to know the
   exact functional path of a PW between PEs. This is provided by the
   traceroute service of the underlying PSN. The opaque nature of the
   PW means that this traceroute information is only available within
   the provider network; e.g., at the PEs.

9. IANA Considerations

   IANA considerations will be identified in the PWE3 documents that
   define the PWE3 encapsulation, control, and management protocols.

10. Security Considerations

   PWE3 provides no means of protecting the integrity, confidentiality,
   or delivery of the native data units. The use of PWE3 can therefore
   expose a particular environment to additional security threats.
   Assumptions that might be appropriate when all communicating systems
   are interconnected via a point-to-point or circuit-switched network
   may no longer hold when they are interconnected with an emulated wire
   carried over some types of PSN. It is outside the scope of this
   specification to fully analyze and review the risks of PWE3,
   particularly as these risks will depend on the PSN. An example
   should make the concern clear. A number of IETF standards employ
   relatively weak security mechanisms when communicating nodes are
   expected to be connected to the same local area network. The Virtual
   Router Redundancy Protocol [RFC3768] is one instance. The relatively
   weak security mechanisms represent a greater vulnerability in an
   emulated Ethernet connected via a PW.

   Exploitation of vulnerabilities from within the PSN may be directed
   to the PW Tunnel end point so that PW Demultiplexer and PSN tunnel
   services are disrupted. Controlling PSN access to the PW Tunnel end
   point is one way to protect against this. By restricting PW Tunnel
   end point access to legitimate remote PE sources of traffic, the PE
   may reject traffic that would interfere with the PW Demultiplexing
   and PSN tunnel services.

   Protection mechanisms must also address the spoofing of tunneled PW
   data. The validation of traffic addressed to the PW Demultiplexer
   end-point is paramount in ensuring integrity of PW encapsulation.
   Security protocols such as IPSec [RFC2401] may be used by the PW
   Demultiplexer Layer in order provide authentication and data
   integrity of the data between the PW Demultiplexer End-points.

   IPSec may provide authentication, integrity, and confidentiality, of
   data transferred between two PEs. It cannot provide the equivalent
   services to the native service.

   Based on the type of data being transferred, the PW may indicate to
   the PW Demultiplexer Layer that enhanced security services are
   required. The PW Demultiplexer Layer may define multiple protection
   profiles based on the requirements of the PW emulated service. CE-
   to-CE signaling and control events emulated by the PW and some data
   types may require additional protection mechanisms. Alternatively,

the PW Demultiplexer Layer may use peer authentication for every PSN
packet to prevent spoofed native data units from being sent to the
destination CE.

The unlimited transformation capability of the NSP may be perceived
as a security risk. In practice the type of operation that the NSP
may perform will be limited to those that have been implemented in
the data path. A PE designed and managed to best current practice
will have controls in place that protect and validate its
configuration, and these will be sufficient to ensure that the NSP
behaves as expected.

11. Acknowledgements

We thank Sasha Vainshtein for his work on Native Service Processing
and advice on bit stream over PW services and Thomas K. Johnson for
his work on the background and motivation for PWs.

We also thank Ron Bonica, Stephen Casner, Durai Chinnaiah, Jayakumar
Jayakumar, Ghassem Koleyni, Danny McPherson, Eric Rosen, John
Rutemiller, Scott Wainner, and David Zelig for their comments and
contributions.

12. References

12.1. Normative References

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

   [RFC768]   Postel, J., "User Datagram Protocol", STD 6, RFC 768,
                August 1980.

   [RFC2401]   Kent, S. and R. Atkinson, "Security Architecture for the
                Internet Protocol", RFC 2401, November 1998.

   [RFC2474]   Nichols, K., Blake, S., Baker, F., and D. Black,
                "Definition of the Differentiated Services Field (DS
                Field) in the IPv4 and IPv6 Headers", RFC 2474, December
                1998.

   [RFC3592]   Tesink, K., "Definitions of Managed Objects for the
                Synchronous Optical Network/Synchronous Digital Hierarchy
                (SONET/SDH) Interface Type", RFC 3592, September 2003.

   [RFC2661]   Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn,
               G., and B. Palter, "Layer Two Tunneling Protocol "L2TP"",
               RFC 2661, August 1999.

   [RFC2784]   Farinacci, D., Li, T., Hanks, S., Meyer, D., and P.
               Traina, "Generic Routing Encapsulation (GRE)", RFC 2784,
               March 2000.

   [RFC2890]   Dommety, G., "Key and Sequence Number Extensions to GRE",
               RFC 2890, September 2000.

   [RFC3032]   Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y.,
               Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack
               Encoding", RFC 3032, January 2001.

   [RFC3550]   Schulzrinne, H., Casner, S., Frederick, R., and V.
               Jacobson, "RTP: A Transport Protocol for Real-Time
               Applications", STD 64, RFC 3550, July 2003.

12.2. Informative References

   [DVB]   EN 300 744 Digital Video Broadcasting (DVB); Framing
               structure, channel coding and modulation for digital
               terrestrial television (DVB-T), European
               Telecommunications Standards Institute (ETSI).

   [RFC3815]   Cucchiara, J., Sjostrand, H., and J. Luciani,
               "Definitions of Managed Objects for the Multiprotocol
               Label Switching (MPLS), Label Distribution Protocol
               (LDP)", RFC 3815, June 2004.

   [RFC3813]   Srinivasan, C., Viswanathan, A., and T. Nadeau,
               "Multiprotocol Label Switching (MPLS) Label Switching
               Router (LSR) Management Information Base (MIB)", RFC
               3813, June 2004.

   [MPLSIP]   Rosen et al, "Encapsulating MPLS in IP or Generic Routing
               Encapsulation (GRE)", Work in Progress, March 2004.

   [RFC1191]   Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191,
               November 1990.

   [RFC1958]   Carpenter, B., "Architectural Principles of the
               Internet", RFC 1958, June 1996.

   [RFC1981]   McCann, J., Deering, S., and J. Mogul, "Path MTU
               Discovery for IP version 6", RFC 1981, August 1996.

   [RFC2022]   Armitage, G., "Support for Multicast over UNI 3.0/3.1
                based ATM Networks", RFC 2022, November 1996.

   [RFC3768]   Hinden, R., "Virtual Router Redundancy Protocol (VRRP)",
                RFC 3768, April 2004.

   [RFC3022]   Srisuresh, P. and K. Egevang, "Traditional IP Network
                Address Translator (Traditional NAT)", RFC 3022, January
                2001.

   [RFC3448]   Handley, M., Floyd, S., Padhye, J., and J. Widmer, "TCP
                Friendly Rate Control (TFRC): Protocol Specification",
                RFC 3448, January 2003.

   [RFC3812]   Srinivasan, C., Viswanathan, A., and T. Nadeau,
                "Multiprotocol Label Switching (MPLS) Traffic Engineering
                (TE) Management Information Base (MIB)", RFC 3812, June
                2004.

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

13. Co-Authors

   The following are co-authors of this document:

   Thomas K. Johnson
   Litchfield Communications

   Kireeti Kompella
   Juniper Networks, Inc.

   Andrew G. Malis
   Tellabs

   Thomas D. Nadeau
   Cisco Systems

   Tricci So
   Caspian Networks

   W. Mark Townsley
   Cisco Systems

Craig White
Level 3 Communications, LLC.

Lloyd Wood
Cisco Systems

14. Editors' Addresses

Stewart Bryant
Cisco Systems
250, Longwater
Green Park
Reading, RG2 6GB,
United Kingdom

EMail: stbryant@cisco.com

Prayson Pate
Overture Networks, Inc.
507 Airport Boulevard
Morrisville, NC, USA 27560

EMail: prayson.pate@overturenetworks.com