Network Working Group
Internet Draft
Expiration Date: December 2005

Ping Pan
(Hammerhead Systems)

July 2005

Dry-Martini: Supporting Pseudo-wires in Sub-IP Access Networks

draft-pan-pwe3-over-sub-ip-01.txt

Copyright Notice

Abstract

Several recent developments have significantly affected the carrier access networks. First, all large carrier backbones have migrated to MPLS. Second, carriers are upgrading access circuits with less expensive and high-speed Ethernet. Finally, the carriers will be offering advanced data services over the access network infrastructure. Subsequently, the

carriers have to face challenges in migration cost, data transport efficiency and edge-to-edge user traffic management.

The Dry-Martini architecture is designed to help the carriers to alleviate these challenges. It provides an approach to establish and maintain pseudo-wires over any access network infrastructure, while stripping off much of the IP/MPLS routing and signaling features that are irrelevant in access network. As a result, all the existing transport equipment, such as SONET/SDH MSPP, can provide MPLS pseudo-wire functionality without much change to the existing platform. Further, due its simplicity, this approach allows the new access devices, such as PONËs and Ethernet CPEËs, to maintain low cost, while being able to interface with MPLS networks.

This document assumes that the reader has at least some familiarity with MPLS and pseudo-wire technologies.

1. Specification of Requirements

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.

2. Introduction

The challenges in expanding carrier networks are mainly taking place at access. The driver behind the access network expansion is to bring high-speed links toward the end users so that the carriers can offer extensive data services.

Subsequently, the carriers have to face a number of issues:

First of all, how to deal with the existing installation? Traditionally, carrier access networks consist of DS1/3 and OC3/12 circuits supporting services such as Frame Relay and ATM. The networks are constructed with either SONET/SDH rings, and/or leased lines. Most of the customer traffic is carried over circuit interfaces, despite the fact that IP data is becoming the predominating traffic type.

Carrying packets over circuit networks cannot take the advantage of statistical multiplexing gains that has made packet networks very efficient. According to a recent report from AT&T [ATT], the multiplexing gain in data-aware access/metro networks is in the range

of 3-4.

Second, how to manage and control the access networks? Most likely, the carriers would like to adapt a single, coordinated architecture in access, metro and backbone networks. ItËs noteworthy that such architecture does not necessarily imply that all traffic flows will be packetized and individually routed as IP packets. It simply means that it is preferable to have a somewhat consistent control-plane throughout the network. Given the backbone has already migrated to IP/MPLS, naturally, the rest of the network should try to adapt to something similar.

Finally, how to deal with the emerging Metro Ethernet? Access technology normally changes very 8-10 years. Most recently the carrier transport Ethernet technology has become the choice of access in metro network. When examining the previous network access technologies, we will notice that X.86, Frame Relay, ATM and Ethernet all share the same set of properties at time when they were deployed:

(a) More bandwidth: this helps data statistical multiplexing

(b) Robustness: failure detection and recovery are critical

(c) Traffic assurance: user flows need to be shaped and policed

(d) Cheap: this is the essential element for wide deployment

Hence, this poses the question: is it possible to engineer the access network in such as a way that they donËt have to be re-engineered very so often?

One way to solve all the above issues is to replace the entire network equipment (including those in access and metro area) with MPLS routers, and interconnect them with big Ethernet pipes. The problem with this approach is in its willful ignorance of carrier economics. Unless it is green-field deployment, we believe that all carriers will have to focus on migration and operation cost as they expand data-friendly access network, and leverage the existing access infrastructure as much as possible.

In this document, we will present the Dry-Martini architecture that provides a solution that is transparent to Layer-2 access technologies and can aggregate any type of data traffic from access into metro core.

Before the description of the architecture, we first evaluate some of the important components, and gain a better understanding on what is and is not important in this area of the network.

3. Components of the Architecture

The figure below illustrates various relevant components in the architecture.

```
                |                          |
                |<-- label negotiation -->|
             +--------+              +----------+    +----------+
 User-1 -----| Access |              |  Metro   |    |   MPLS   |
             | Device |--------------|Aggregator|----| Backbone |-...
 User-2 -----|        |   sub-IP     |          |    |  Router  |
             +--------+ data trunks +----------+    +----------+
                |                          |
                |<----- Pseudo-wires ---->|
```

Figure-1: A simplistic view of the carrier network

3.1. Access Devices and Data Trunks

Traditionally, a typical Access Device can be a circuit aggregation switch (e.g. CPE), or a SONET/SDH switch (e.g. MSPP and ADM). The data trunks are normally DS1/DS3 circuits running Frame Relay and ATM, or OC-3/OC-12 circuits running ATM. At Access Device, each user flow is mapped to one of the circuits.

In recent years, the carriers have begun to deploy high-speed Ethernet circuits and lambda (photonic) as data trunks to the customers. Access Devices have evolved as well. The PONs (Passive Optical Networks) in various flavor have been evaluated and deployed for high-speed network access.

The links that interconnect the access devices and aggregators can be (and not limited to) lambda (photonic), SONET/SDH cross-connects, ATM/Frame Relay circuits, or Ethernet VLAN connections. For clarity, we denote these links as sub-IP data trunks, the networks that setup and maintain sub-IP data trunks, as sub-IP networks.

It is important to realize that, it is carrierËs interest to aggregate as many user flows into a sub-IP data trunk as the SLA would allow, in taking the advantage of statistical multiplexing gains.

To support wide deployment, the Access Device must be reliable, simple-to-manage and cheap. Normally, each access device connects to the metro backbone through two links for protection purpose. Extensive IP routing functionality at access devices is not always necessary.

3.2. Metro Aggregators

Depending on metro network topology, each aggregator may process traffic from dozens to hundreds of access devices. Upon grooming user traffic together, the aggregators feed the traffic toward the MPLS core.

Typically, each aggregator can be a combination of multiple systems. They are responsible for multiplexing and de-multiplexing DS3/1/0 or OC-n circuits, and switching data packets. As the metro network becomes more data-centric, the aggregators are expected to be more data friendly.

Some of the key requirements in supporting data traffic on aggregators are per-user-flow QoS, data trunk OAM, and protection and recovery in event of both equipment and facility failures.

3.3. Packet Flow ID

Each user flow can be uniquely identified with a packet flow ID. Depending on the transport technology, the flow ID can be (and not limited to) Frame Relay DLCI, ATM VPI/VCI, Ethernet VLAN (including VLAN tag via Q-in-Q), and MPLS Label.

Using packet flow IDs enables the carriers to offer logical circuits to the customers, and, thereby, manage and control user traffic at per-flow granularity.

3.4. Pseudo-wire Encapsulation

Aggregating multiple flows into a shared data trunk requires packet flow ID encapsulation. There have been a number of encapsulation methods for various technologies, such as Q-in-Q for Ethernet.

Draft-martini, known for its original designer Luca Martini, has presented an encapsulation method that enables the aggregation of

different types of Layer-2 flows into a single trunk, while retain some of the important service-specific characteristics ([PWE3-ETHER], [MARTINI-ATM], [MARTINI-FR]). Essentially it is to encapsulate the packets belong to a user flow with a MPLS header and a Layer-2 specific control word. The MPLS label [RFC3032] in the header makes the flow unique throughout the network. Upon entering the network, each encapsulated flow is referred as a Pseudo-wire÷.

Originally, draft-martini was designed to help the carriers to aggregate Layer-2 traffic over a common MPLS/IP backbone via MPLS Label Edge Routers (LERËs). Each Layer-2 flow is mapped to a Pseudo-wire. The setup and maintenance of each Pseudo-wire involve two provider edge nodes (PEËs), which exchange connection and encapsulation label information through targeted LDP [PWE3-CTRL].

Upon closer evaluation, we note the following: one of the most important benefits in Pseudo-wire is that the carrier can handle and process user traffic at per-flow granularity independent of the underlying networking technology. On the contrary, encapsulation mechanisms, such as Ethernet Q-in-Q, will only operate in a specific Layer-2 environment. We will elaborate this point later on.

3.5. Label Negotiation

Pseudo-wires operate between two network nodes. Each Pseudo-wire consists of two unidirectional paths, one in each direction. Each path can be uniquely identified by the triple <node-1, node-2, encapsulation-label>. Prior to data transmission, two nodes need to know the value of the encapsulation-label for each user flow.

As mentioned above, the most common label negotiation mechanism is target LDP [LDP, PWE3-CTRL], where two edge nodes can initiate a LDP session and exchange label binding information to setup the Pseudo-wires.

The actual Pseudo-wire setup sequence is very simple. However, the overhead in processing LDP protocol itself, such as LDP session and adjacencies management and peer discovery can be quite elaborate. This is because the original LDP specification was designed to establish MPLS LSPËs among routers in the backbone environment. Hence, applying the same label negotiation mechanism in access network may be an over-kill. We will discuss this issue later on.

3.6. Traffic Policing and Assurance

Most of the data applications do not require stringent QoS. In todayËs backbone networks, the carriers over-provision the network links, and rely on DiffServ to overcome temporary traffic congestion. Thus, per-flow shaping and rate limiting does not apply in enterprise and backbone networks. However, we should not ignore QoS guarantees (or SLA) as an essential part of carrier service offering.

As the access network begins to expand, the mixture of existing low-speed circuit infrastructure and high-speed Ethernet links will cause network resource heterogeneously distributed. As carriers continue to lease service lines to enterprise customers, and offer QoS-sensitive data applications (e.g. voice) to consumer users, supporting per-user-flow QoS becomes increasingly important.

3.7. Demarcation Points and Pseudo-wire Switching

The concept of UNI (User-Network Interface) has been foreign to the Internet, where the entire network supposes to be distributed and open. From user application perspective, this is true: the end-users should never observe the existence of any UNI interface, as elaborately defined in ATM.

Nevertheless, in access/metro area, there remain demarcation points where traffic from one segment of the network is transferred to another accordingly to a set of rules between carriers. For instance, there likely exists a demarcation point between metro network (aggregators) and the backbone (MPLS/IP routers). As such, depending on carrierËs deployment scenario, the Pseudo-wires coming from access networks need to be policed before switching toward the backbone.

Another demarcation point where subsequent Pseudo-wire switching is useful is between two metro networks. Depending on the bilateral (and multilateral) agreement, user flows in the form of Pseudo-wires will be handed off (or switched, stitched) from one network to another.

4. Dry-Martini Architecture

The Dry-Martini Architecture is based on draft-martini as the encapsulation method for aggregating user flows into carrier networks, and strip off much of the IP/MPLS routing and signaling features that are irrelevant in access networks. Because we have relaxed and simplified much of the constraints in the original design, we refer this architecture as dry-martini÷.

Our rational is as follows: the operation of Pseudo-wires involves two endpoints of the connection, and is almost independent from the operation taking place within the underlying network. Hence, we argue that the pseudo-wire concept is applicable in all networks. (Note that much of the concept of supporting Pseudo-wires over any packet networks (PSN) has been assumed in the IETF PWE3 architecture [PWE3-ARCH].)

Further, pseudo-wire encapsulation is transparent to all the existing Layer-2 technologies, and, perhaps, to the access technologies that will get deployed in the future (such as WiMax). Therefore, if we always handle user flows at Pseudo-wire layer, it will provide carriers with a uniformed and consistent method to aggregate, police and manage all types of data flows. This will subsequently simplify carrier service migration and integration.

Another advantage in Pseudo-wires is in its flexibility to map user applications into flows.

For example, today, the carriers can packetize voice streams into Pseudo-wires [SATOP, CESoPSN] and transport them over the backbone. Voice service requirements can be retained with some basic QoS treatment (such as clocking recovery). This implies that, if the carriers can manage data flows as Pseudo-wires with QoS, redundancy and OAM features, then any stream traffic (such as video and voice) sent over Pseudo-wires will always get appropriate edge-to-edge SLA guarantees.

In essence, Pseudo-wire can be the convergence layer for transporting data flows over any network. Figure 2 shows the Pseudo-wire in relation to user applications and underlying transport network.

```
       +-------------------+       user         +-------------------+
       |     Payload       |   applications     |     Payload       |
       | (IP, L2 data, ... |<==================>| (IP, L2 data, ... |
       | packetized voice) |                    | packetized voice) |
       +-------------------+                    +-------------------+
       |                   |     convergence    |                   |
       |                   |        layer       |                   |
       |   Pseudo-wires    |<==================>|    Pseudo Wires   |
       |                   |                    |                   |
       +-------------------+                    +-------------------+
       |     Layer 2       |     data service   |       Layer 2     |
       |(Ethernet, ATM...) |<==================>|(Ethernet, ATM...) |
       |                   |                    |                   |
       +-------------------+                    +-------------------+
       |     Layer 1       | physical transport |       Layer 1     |
       | (optical, TDM...) |<==================>| (optical, TDM...) |
       |                   |                    |                   |
       +-------------------+                    +-------------------+
```

Figure 2: The logical layering model applied
in Dry Martini architecture

Within the Dry-Martini architecture, the carriers can operate Pseudo-wires over any sub-IP networks:

Application 1: The carriers may create Pseudo-wires from SONET/SDH access devices (such as MSPPËs) directly, and aggregate user Ethernet traffic over the existing metro infrastructure. TodayËs typical EoS (Ethernet-over-SONET) solution is to use VCAT and GFP, and map Ethernet physical port to a Virtual Concatenation Group (VCG). With the Dry-Martini architecture, the carriers can map multiple Ethernet flows into a single VCG. This can improve access bandwidth utilization significantly.

Application 2: The carriers may choose to create Pseudo-wires and aggregate data packets over the existing leased ATM or Frame Relay circuits. Once again, improving link utilization (a.k.a. statistical multiplexing gains) may be an important economical factor here.

Application 3: The carriers can always aggregate multiple user flows into a single wavelength off the PONËs, and process the flows at aggregators.

In all the applications, carriers use the method of their choice to

manage the sub-IP data trunks: GMPLS for optical networks, PNNI/SPVC for ATM infrastructure, Ethernet signaling for metro Ethernet, or, simply, static configuration for SONET/SDH connection provisioning. The key is that, as long as there is an operational sub-IP data trunk between two network nodes, the carrier may establish Pseudo-wires to aggregate data flows over it.

In the architecture, the access devices do not need to support much of the IP functionality, such as per-packet IP routing, and routing protocols. All data flows are handled as circuit-like Pseudo-wires. Given the access devices have only a couple of connections to the metro backbone, the use of IP routing would be very limited. It is the aggregators that need to be able to interface with both access networks and the MPLS/IP network, and aggregate and switch user traffic in between.

In the remaining of the document, we will outline the data-plane and control-plane issues in supporting the Dry-Martini architecture, and articulate some of the important features that are actually needed in this area of the network.

5. Data-Plane Operation in the Dry-Martini architecture

As an example of the operation, we consider the network setup shown in Figure-1. Suppose that there are N user flows arriving at the Access Device. The user flows are provisioned prior to the actual data transmission. A typical user flow may be a privately leased line for an enterprise customer, or a high-speed Ethernet connection to a residential location. Thus, we assume that all the user flows have a relatively long duration, and each flow may have some QoS parameters (such as average rate) associate with it.

Upon the reception of a packet from a user, the access device will first search for the corresponding flow information (such as the VLAN tag) from its local cached database. If a match is found, it will run a simple CAC (Call Admission Control) algorithm to ensure QoS compliance and encapsulate the packet with a new packet flow ID, and then transmit the packet toward the aggregator.

The packet flow ID is negotiated with the aggregator ahead of time. Multiple user flows can share a common data trunk with different flow IDËs.

At the aggregator, the packet flow ID will be examined and stripped off. The packets will then be forwarded toward the core.

The packet forwarding sequence from the aggregator to the access device would be same.

All packet encapsulation should be the same as the ones defined in draft-martini. However, we do not mandate the MPLS label as the only packet flow ID. Depending on the flexibility of the access device itself, the packet flow ID can be something different, for instance, Ethernet VLAN tag.

Ethernet VLAN can be used as the Pseudo-wire flow ID between access devices and aggregators. This is reasonable for the following reasons: first, given the network size between access devices and aggregators, VLAN scaling (that is, 4000 VALN tags per interface) should not a real problem. Second, some of the access devices may not have the ability to support various MPLS label manipulation (push, pop and stack) operations.

In summary, the data-plane requirement on ant access device is very trivial. No per-packet routing lookup is required. However, the access device needs to be able to police user traffic on per-flow basis.

The procedure on the aggregator is similar, expect that the aggregators interface with the core, and may need to exercise more extensive packet processing functions with the core routers.

6. Control-Plane in the Dry-Martini architecture

As mentioned before, label negotiation takes place between the access devices and the aggregators.

In practice, there are two general approaches in setting up Pseudo-wire labels: in-band and out-band. The in-band approach is to exchange control messages over the sub-IP data trunks. The out-band approach is to setup labels through an external management system.

There are a number of ways to exchange IP control messages (e.g., LDP messages) between the edge nodes. One approach is to route the messages through the underlying sub-IP network. For example, in SONET/SDH networks, the control messages may utilize DCC channels to communication with each other. However, this would require every node in the sub-IP network to be IP-capable, which may be not practical in many of the operational networks.

We propose to "tunnel" all control packets through the sub-IP data trunks as regular data payload from the edge. The advantage here is

that the exchange of control messages will not disturb the operation in the underlying sub-IP network.

For the user packets to be encapsulated with a MPLS header, we require control packets to be encapsulated with "IP4 Explicit NULL Label" [RFC2032]. At the destination, the label is popped, and the control packets are delivered to the control plane for further processing.

For the user packets to be encapsulated with a VLAN tag, we propose to use a special VLAN value for control message delivery. Once again, at the destination, the messages are picked up for further examination.

6.1. Option 1: Target LDP

The conventional method is to run target LDP in-band between access device and aggregator. For clarity, we repeat the procedure described in [PWE3-CTRL] in the context of setting up Pseudo-wires in sub-IP networks.

Each Pseudo-wire consists of two unidirectional paths, one in each direction. The access device and the aggregator are the two edge nodes. Each edge initiates the setup of the path on behalf of ingress Layer-2 traffic. Each path is uniquely identified by the triple <PE-1, PE-2, VCID>, where the VCID is a 32-bit quantity that must be unique in the context of a single LDP session between two edges. For a given Pseudo-wire, the same VCID must be used when setting up both paths.

In this case, the access device needs to implement target LDP to communicate with the aggregator.

6.2. Option 2: Lightweight Signaling

In certain scenarios, using target LDP for Pseudo-wire label negotiation is questionable. LetËs first evaluate the tradeoffs in supporting target LDP in this part of the network.

As mentioned before, much of the LDP functionality is irrelevant in setting up Pseudo-wires. Its built-in discovery and session and adjacencies management algorithms are originally designed to operate in a distributed networking environment and interface among routers.

However, in access networking environment, the network configuration is most likely a simple spoke topology. In most cases, the access devices and the aggregators are one-hop away.

From the operation point of view, the carriers would like to control the label allocation and distribution at the aggregators, rather than from some customer-location access devices. The actual Pseudo-wire negotiation signaling should be more client-server style, instead of peer-to-peer as in LDP.

Further, letËs examine the development and performance cost. One of the primary requirements in this part of the network is low-cost. Supporting MPLS signaling and IP routing protocols will no doubt require more expensive components in developing the access devices. Typically, each metro aggregator may interface with hundreds of access devices. Supporting target LDP implies that each aggregator may need to maintain hundreds of TCP and LDP sessions at control-plane. This adds unnecessary performance overhead to the aggregators.

We believe that there is a need for developing a light-weight Pseudo-wire negotiation signaling protocol for access networks. Some of the key elements of the protocols are:

(a) Client-server signaling: If we look at the operation of Frame Relay LMI, in practice, the DTE-DCE relationship is nearly identical to that of access devices and metro aggregators.

(b) Light-weight: The complexity of the protocol should be similar to that of ICMP. In other words, every access device should be able to support it without much cost associated with it.

(c) In-band: This will increase the automation capability during Pseudo-wire setup. Further, in-band protocols can always help the network nodes in failure detection.

The detail design of the lightweight protocol is beyond the scope of this architecture document. We will provide its design elsewhere.

In summary, using a lightweight protocol would enable the access devices to setup pseudo-wires without dealing with IP routing and full-stack MPLS signaling. When designed correctly, the aggregators would have an efficient control-plane interface with the access devices.

6.3. Option 3: Pseudo-wire Proxy

This method is to deploy proxies÷ to manage Pseudo-wire allocation and distribution. Figure 3 shows its configuration. The Pseudo-wire proxy is a logical entity that can operate in carrierËs NOC, or on metro aggregators.

```
                       +-----------------+
                   +---|Pseudo-wire Proxy|---+
                   |   +-----------------+   |
                   |                         |
                +--------+             +----------+     +----------+
    User-1 -----| Access |             |  Metro   |     |   MPLS   |
                | Device |-------------|Aggregator|-----| Backbone |
    User-2 -----|        |   sub-IP    |          |     |  Router  |
                +--------+ data trunks +----------+     +----------+
                   |                         |
                   |<--------- OAM --------->|
```

Figure-3: Pseudo-wire management with a Pseudo-Wire Proxy

The operation sequence can be as follows: a customer negotiates with the carrier on his network access service, which may include an extensive set of business and technical conditions. During the negotiation, the carrier knows where the customerËs traffic will enter the network, and the access devices, the aggregators and the data trunks to be used. From the proxy, the carrier assigns the Pseudo-wire labels to both the access device and the aggregator through the management interface (e.g. SNMP).

This type of out-band, static Pseudo-wire configuration is simple to implement. However, without some type of direct communication between the access devices and the aggregators, any failure in the data trunk will not be detected, or too late to be notified. Hence, we recommend enabling the Pseudo-wire OAM functionality when operate in this mode.

7. Carrier Deployment Considerations

7.1. Pseudo-wire Switching

As shown in Figure 1, the aggregators interface with the access devices via pseudo-wires, and can interface with the core routers with MPLS or Pseudo-wires. By switching pseudo-wires between the access networks and the metro core, the carriers would have the ability to control the user flows edge-to-edge.

Much of the work in Pseudo-wire switching is taking place in IETF at the moment. We will not expand the description any further at this point.

8. Security Considerations

This document extends the use of PWE3 to sub-IP networks. It has the same security requirements as in PWE3.

9. Contributors

Tad Hofmeister, Tedi Tedijianto and Calcin Leung have contributed to the original dry-martini ideas back in the fall of 2002. Much of the ideas and concepts have been thoroughly discussed and validated with a number of my colleagues in Hammerhead Systems and operation and architecture folks in various carriers.

10. Normative Reference

[ATT] T. Afferton, et al, "Packet Aware Transport for Metro Networks", IEEE Network Magazine, April 2004.

[RFC3032] E. Rosen, et al, "MPLS Label Stack Encoding".

[PW-CTRL] L. Martini, et al, "Pseudo-wire Setup and Maintenance using LDP", draft-ietf-pwe3-control-protocol-14.txt

[LDP] L. Andersson, et al, "LDP Specification", draft-ietf-mpls-rfc3036bis-00.txt

[PWE3-ARCH] S. Bryant, P. Pate, "PWE3 Architecture", draft-ietf-

pwe3-arch

[PWE3-ETHER] L. Martini, et al, "Encapsulation Methods for Transport of Ethernet Frames over IP/MPLS Networks", draft-ietf-pwe3-ethernet-encap

[MARTINI-ATM] L. Martini, et al, "Encapsulation Methods for Transport of ATM Cells/Frame over IP and MPLS Networks", draft-martini-atm-encap-mpls

[MARTINI-FR] C. Kawa, et al, "Frame Relay Encapsulation over Pseudo-wires", draft-martini-frame-encap-mpls

[CESoPSN] A. Vainshtein, et al, ?Structure-aware TDM Circuit Emulation Service over Packet Switched Network?, IETF Draft

11. Informative References

[PWE3-TRANSPORT] L. Martini, et al, "Transport of Layer 2 Frames over MPLS", draft-martini-l2circuit-trans-mpls

[SAToP] A. Vainshtein, Y. Stein, ?Structure-Agnostic TDM over Packet?, IETF Draft

[TDMoIP] Y. Stein, et al, ?TDM over IP?, IETF Draft

[CEP] A. Malis, ?SONET/SDH Circuit Emulation over Packet?, IETF Draft

12. Author Information

Ping Pan
Hammerhead Systems
640 Clyde Court
Mountain View, CA 94043
e-mail: ppan@hammerheadsystems.com

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

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