Internet Draft
Expires February 2005

Peter Willis et al
BT
September 2004

Service Provider requirements for PWs
draft-willis-pwe3-requirements-00.txt

Status of this Memo

Abstract

This internet draft provides some requirements to help steer future PWE3 work from the perspective of a Service Provider.

1. Introduction

This document, although not exhaustive, captures some of the more important requirements to be met when considering further work on Pseudo-Wires (PW), such as PW stitching. This draft is NOT aimed at modifying existing PWE3 encapsulations but is aimed at providing guidance to future PWE3 work e.g. PW stitching.

Note on terminology used:

This document uses the more generic term "client" to refer to the PWE3 payload and the more generic term "server" to refer to the PSN. It is commonly understood (by software developers and network architects) that client/server relationships can be recursive so the terms "client" and "server" are used in this document to avoid the need to enumerate all client/server stacks fully, as would be the case if we used the terms "payload" and "PSN". It should be noted that client/server recursion is a fundamental requirement for Service Providers (SP) and not just an architectural possibility. For example SP A may buy connectivity from SP B and also sell connectivity to SP C (In this example A,B,C have client/server relationships and are NOT peers). SP C may sell connectivity to an Enterprise. Even in this simple example we have 4 recursions of a client/server relationship and 4 respective layer networks.

1.1 PWE3 Payload/PSN Independence Relationships

PWs create a client/server relationship between 2 layer networks. There are two types of client/server relationships that must be considered:

Case 1 - where the client and server layer networks are owned by the same Service Provider (SP).

Case 2 - where the client layer network is owned by a customer, who may be a SP (SP A), and the server layer network is owned by a (different) SP (SP B).

For case 2, the functional components (such as routing, signalling, OAM, management etc.) of the client layer network must be completely independent of the functional components of the server layer network. Although it is possible to design solutions where the server layer network's functional components interact with the client layer network's functional components this approach leads to the following undesirable consequences:

1. **The service may break if the client changes any of their functional components.**
2. **The SP has to track developments in the clients' technology** and implement upgrades in their network accordingly.
3. **Under fault conditions it becomes difficult to establish if** the fault is in the client or server network.

By requiring that the client and server layers are able to be run

independently of one another, it naturally follows that the server

layer should transparently transfer the client layer. For example consider the case where the client is an ATM network. The client may implement a proprietary feature (e.g. AAL, non-PNNI routing and signalling, OAM) which if not carried transparently would break the service.

This is not only a technical requirement but it also has commercial implications because a SP is likely to consider the details of their network to be commercially sensitive and will therefore wish to hide those details from any client layer networks. For example it would not be desirable for the server layer network to peer with the routing & signalling of the client in the example above.

Where the client and server layer networks are owned and operated by the same SP (case 1 above) it may be possible to relax the degree of independence between client and server layers. However, the server layer should still be able to transparently transfer the client layer network's data-plane. Non-transparent transfer optimisations may save a tiny amount of bandwidth and possibly improve routing/signalling protocol scalability but they ultimately increase operational complexity, eg the behaviour for each spin of client compression is different and requires a case by case consideration.

1.2 Basic OAM requirements

The transport defects of a layer network are determined by its mode. There are 3 basic modes:

1 **Connectionless packet switching (CL-PS) examples are IP and** Ethernet.
2 **Connection Oriented Packet Switching (CO-PS) examples are Frame** Relay, ATM and MPLS (RSVP-TE).
3 **Connection Oriented Circuit Switching (CO-CS) examples are SDH/SONET** and optical wavelength switching networks.

It should be noted that multi-point to point (MP2P) LDP creates a type of MPLS that is a special case of the CO-PS mode. The MP2P LDP mode is widely implemented today and its requirements are also addressed in this document.

The transport defects can be summarized as follows:

Connectionless Packet Switching

(i) Breaks

Connection Oriented Circuit Switching (co-cs)

(i) Breaks
(ii) Swaps, but only between exactly alike entities (e.g. an SDH VC4

can swap with another SDH VC4 but not with another SDH VC12.

Alternatively a wavelength with colour A can swap with another wavelength of colour A but not with a wavelength of colour B.)
This is because the link-connections identifiers are constrained to take on a real/physical appearance in either time/freq/space

Connection Oriented Packet Switching (co-ps)

(i)   Breaks
(ii) Swaps between any entities
(iii) Mismerge. A mismerge is where traffic from one connection (e.g. LSP, ATM VC) leaks into another connection.
Mismerges have the following subcases (where A1 and B1 are sources, and A2 and B2 are sinks)
- A1->A2 mismerging into B1->B2, with A1->A2 traffic seemingly unaffected
- A1->A2 mismerging into B1->B2, with A1->A2 traffic broken
- A1->A2 self-mismerging back into A1->A2 (One example of a where self-mismerge can occur is due to a routing loop.)

MP2P LDP MPLS

(i)   Breaks. This is a complete failure of the whole MP2P tree.
(ii) Swaps between any entities.
(iii) Mismerges, with the following subcases (where A1 and B1 are sources, and A2 and B2 are sinks)
- A1->A2 mismerging into B1->B2, with A1->A2 traffic seemingly unaffected
- A1->A2 mismerging into B1->B2, with A1->A2 traffic broken
- A1->A2 self-mismerging back into A1->A2 (One example of a where self-mismerge can occur is due to a routing loop.)
(iv) Partial Breaks. It is possible for only partial failure of the MP2P tree topology i.e. only some branches break so only some ingresses are disconnected from the egress.
(v)   Partial swaps. It is possible that only some of the branches of the MP2P tree experience swaps. The result would be that the traffic egressing would be a mix of correct & incorrect traffic.

Many Service Providers operate ECMP on their MP2P LDP MPLS networks. If ECMP is used then there further types of partial breaks & swaps are introduced. This would be where only some of the flows on the MP2P tree experience breaks or swaps. In this case no ingress would be totally disconnected from the egress but some of the flows from some of the ingresses could be disconnected or swapped to an incorrect output.

Suitable OAM must be provided in the traffic data-plane of each mode and the special case of MP2P LDP to automatically detect the above. Defect detection is required to be unidirectional in the co-ps and co-cs modes. Unidirectional detection is required to detect errors in each direction independently i.e. it is possible to unambiguously resolve which direction the defect is operating in. It must also be possible to unambiguously resolve whether the fault is in the control plane or the traffic data plane. For example if the control plane

fails in direction B to A and forwarding A to B continues (because

service providers require data forwarding even when the control plane fails) then the fault must be correctly identified as a control plane fault without false alarms from the data traffic plane OAM.

We must specify appropriate entry/exit criteria and consequent actions for each defect. The entry/exit criteria define when the network service is "up" or "down". It is essential to accurately define network availability for Service Level Agreements based on persistent defects (10s is the normal default here), especially for performance SLAs as performance measures taken whilst the network is "unavailable" should be disregarded (See the "Requirements for SLA verification" section). An example of a "consequent action" would be to suppress the traffic on a connection (LSP, ATM VC) if it is swapped. A consequent action for a break would be to raise an appropriate alarm and may be to initiate a reroute. Generation of forward and backward defect indicators (BDI) would also be a consequent action. BDI should ideally also be supported (this can be in-band or out-of-band) to allow for both direction defect/availability monitoring from a single end.

Further work is required to define all the consequential actions for all the possible defects.

For the co-ps and co-cs modes the OAM must be independent of the manner in which the data-plane path is instantiated, ie whether by signaling (any protocol) or provisioning. If OAM is not independent of the PW instantiation method then not only is operational complexity increased (N types of OAM messages, N MIBs, N fault finding tools) but there is no guarantee the different OAM methods are compatible (e.g. a LDP provisioned PW might be mismerged with a static provisioned PW and the fault may not be detected).

Further, the OAM activation/deactivation must be harmonized with the set-up/tear-down of the path. Failure to harmonize OAM activation/deactivation with PW set-up/tear-down will lead to either:
- lack of OAM protection when the PW is set-up, or false alarms when the PW is torn-down; or
- OAM being activated prior to PW set-up and significant problems due to operator error.

1.3 Client/server OAM requirements

Defects must be detected/handled at the path (co-cs and co-ps) or flow (cl-ps) termination point of a layer network. Failure to do this will lead to ambiguous fault indications which significantly increase operational complexity and the time taken to resolve a fault (e.g. when a fault happens we should avoid passing trouble tickets between SPs to locate the fault - defect detection at the correct termination point of a layer network will aid this).

To prevent alarm storms in any co-cs or co-ps client layer networks a FDI (Forward Defect Indication) signal should be passed to the client layer networks. This must use the appropriate FDI syntax of the OAM

used by the particular client layer technology affected.

1.4 Client/server adaptation requirements

Service Providers who deploy MPLS networks wish to obtain maximum benefit from their MPLS network. If the PWE3 functions assume IP and MPLS are the same then the SP using MPLS gains less benefit from their MPLS network than is possible. By recognizing that MPLS and IP PSNs are different then it should be possible to optimize the PWE3 functions for MPLS which may give benefits. This section discusses the general case of this by considering some of the 9 possible client/server combinations between the 3 network modes of co-cs, co-ps and cl-ps.

The adaptation between a given client and server layer network should be a function of the nature of both the client mode and the server mode, and it is not the same in all cases.

To fully address each one of the possible 9 client/server modal combinations would be an onerous task (noting that we would also need to consider each particular technology as there are some differences at this level too). However, some examples of the issues that need considering for client/server adaptation are given in the rest of this section.

When there is a 'many to one' relationship between the client and the server, and the server is either co-ps or co-cs, then the adaptation function must include a muxing capability. This capability is not required if the server is cl-ps (for any client mode) since the cl-ps layer network does muxing as a consequence of its intrinsic nature. I.e. each packet here carries a network-unique DA/SA pair and client layer identification is carried out via the 'protocol' or 'next protocol' field.

In the case of client/server relationships between the cl-ps and co-ps modes (either way round) the adaptation function may require a fragmentation/reassembly capability if the server layer packet MTU is less than the client packet size.

A fragmentation/reassembly function is clearly not required when the server layer is co-cs for any client layer mode. However, it does require the server layer payload bit rate to exceed the average client layer bit rate, but this is a general traffic requirement anyway for a co-cs server layer. Further, if the client is cl-ps or co-ps, it also implies a need for rate-decoupling (ie idle-fill) and client traffic unit delineation.

As a final example, when the server layer is cl-ps then this may create misordering of the client (any mode). Re-ordering should never occur in a co-ps or co-cs server layer of course for any client mode.

So as we can see, the client/server adaptation cannot be the same for all cases and each pair of modes that form a client/server relationship must be considered in their own right.

In PWE3 terminology this means that the PWE3 processing functions have

no need to be the same for IP and MPLS PSNs. We need to understand the optimum benefits from reusing IP and MPLS PWE3 functions against optimizing the PWE3 functions for the particular PSN that is deployed.

1.5 Requirement for OOB management/control

The sensitive internal control/management-plane protocols must be made secure from attack, and the network must remain stable under situations of extreme stress, i.e. serious failures.

It is therefore a requirement that such protocols should be separated, in terms of performance, addressing and availability (fate sharing), from the customer traffic where this is possible.

It is noted that running the control/management-plane protocols OOB (Out-Of-Band) in relation to customer traffic is an excellent way to satisfy this requirement. OOB may be logical or physical separation. It is possible that current state of the art methods of control plane separation e.g. separate queues for control plane protocols operating in an address space separated from the customer traffic, might satisfy the security requirement but this will be dependent upon implementation detail.

A key implication of the use of an OOB control plane is that the control messages are no longer a reliable means of determining the integrity to the user's data as the control plane traffic and the traffic data plane traffic are subject different failure mechanisms. For example, it cannot be assumed that a failure of the control plane will mean that there is also a failure in the traffic data plane and vice versa. It is highly desirable, and normal operational practice in current connection oriented networks, that any failure in the control plane does not force a failure on the traffic data plane if the traffic data plane is otherwise working correctly. If an OOB control plane is to have reliable information on the state of the traffic data plane, then the traffic data plane must have an in-band OAM flow in order to verify the state of the traffic data plane and pass this state information to the OOB control plane.

1.6 Requirements for SLA verification

SLAs are an increasingly important issue for a SP.

SLAs exist between a SP and a customer (where the customer may be another SP). And in the case of SLAs between different SPs this can be both in a client/server sense (ie SP A leases capacity from SP B) or in a peer-partition sense across an E-NNI between two SP domains in the same layer network.

Note - It is a networking truth that a link-connection (i.e. 'hop') in a client layer network is provided by an end-end path/flow in a server layer network.

The performance of a client layer network is therefore determined by

the performance of itself and that inherited from its server layer.

This is a recursive behaviour to the duct and is something that should be taken into account when considering SLAs.

Although there are 9 possible client/server combinations between the 3 network modes of co-cs, co-ps and cl-ps, some may make more sense than other from performance inheritance and SLA considerations.

This also raises questions regarding end-end performance allocations and their fair apportionment to SP domains. Although this topic is not covered here, it is mentioned because it has significant commercial importance. The network architecture therefore requires careful consideration regarding its ability to allow such a specification and measurement.

A service provider can simplify its SLA monitoring by monitoring at only the layer network(s) that offer service(s) to customers rather than all its server layers. However SLA measuring is not the same as OAM for defect detection/handling. OAM is a generic requirement for all layer networks and is an enabler to SLA monitoring.
SLAs have 2 distinct parts:

- There is an availability part that defines the amount of time (usually as a percentage) that the service must be in the up-state, and

- there is a Network Performance part that defines the transfer metrics/objectives that must be met whilst the service is in the up-state.

There is a logical ordering of processing required to ensure that SLAs are implemented in the correct and most efficient manner.

Firstly, we must respect the allowed connectivity constraints for the mode considered. This is vital so that we can correctly identify the defect types that are relevant. Failure to identify all the ways a network can break will lead to situations where the Operations instrumentation will say a service is working whilst the customers are arguing the service is broken.

Secondly, we must define suitable OAM techniques for the data-plane of the mode considered. This must define appropriate defect entry/exit criteria and consequent actions.

Thirdly, based on defect persistency (usually 10s) we must define the unavailable state entry/exit criteria and consequent actions.

Fourthly, once we have defined/specified the defects and unavailability we now have a temporal basis against which we can define/measure the up-state Network Performance SLA.

Note also that for many services/applications, availability must be a bi-directional parameter. That is, even if only one direction is broken the total service (ie both directions) is considered broken.

This has implications for the starting/stopping of the collecting of

Network Performance measurements for the up-state SLA (noting that during the up-state Network Performance is a unidirectional measurement).

2. Security Considerations

This document raises no security issues per se. However, it does make reference to using techniques to ensure that both the traffic data-plane and the internal/sensitive control/management-plane protocols have security measures in place. For example, two key requirements identified in this document are:

- A trail termination source identifier should be used in the OAM of co-ps and co-cs trails to detect instances of misconnectivity;

- A physical or logical OOB control/management-plane network should be used.

3. Acknowledgements

Many BT folks have contributed in one way or another to this document including: Neil Harrison, Peter Willis, Alan McGuire, Richard Spencer, Ben Niven-Jenkins, Andy Reid, Dave Milham, Tony Flavin, Adrian Smith.

4. Author's contact details

Peter Willis
BT Group CTO
peter.j.willis@bt.com

5. Full Copyright Statement