VLL Services

In This Chapter

This chapter provides information about Virtual Leased Line (VLL) services and implementation notes.

Topics in this chapter include:

ATM VLL (Apipe) Services

This section provides information about the Apipe service and implementation notes.

This feature is supported on the 7450 ESS platform in mixed-mode.

Topics in this section include:

ATM VLL For End-to-End ATM Service

ATM VLLs (Apipe) provide a point-to-point ATM service between users connected to SR nodes on an IP/MPLS network. Users are either directly connected to a PE or through an ATM access network. In both cases, an ATM PVC (for example, a virtual channel (VC) or a virtual path (VP)) is configured on the PE. This feature supports local cross-connecting when users are attached to the same PE node. VPI/VCI translation is supported in the ATM VLL.

PE1, PE2, and PE3 receive standard UNI/NNI cells on the ATM Service Access Point (SAP) that are then encapsulated into a pseudowire packet using the N:1 cell mode encapsulation or AAL5 SDU mode encapsulation according to RFC 4717, *Encapsulation Methods for Transport of ATM Over MPLS Networks*. When using N:1 cell mode encapsulation, cell concatenation into a pseudowire packet is supported. In this application, the setup of both VC and VP level connections are supported.

The ATM pseudowire is initiated using Targeted LDP (TLDP) signaling as specified in RFC 4447, *Pseudowire Setup and Maintenance using LDP*. The SDP can be an MPLS or a GRE type.

Figure 1 shows an example of ATM VLL for end-to-end ATM service.

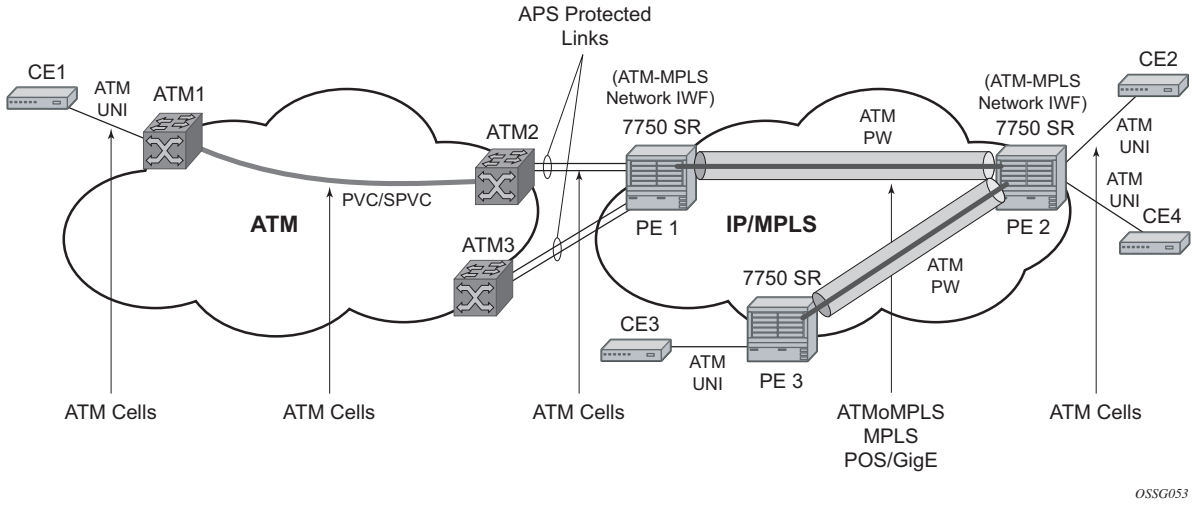


Figure 1: ATM VLL for End-to-End ATM Service

ATM Virtual Trunk Over IP/MPLS Packet-Switched Network

ATM virtual trunk (VT) implements a transparent trunking of user and control traffic between two ATM switches over an ATM pseudowire. Figure 2 depicts ATM 2 and ATM 3 switches that appear as if they are directly connected over an ATM link. Control traffic includes PNNI signaling and routing traffic.

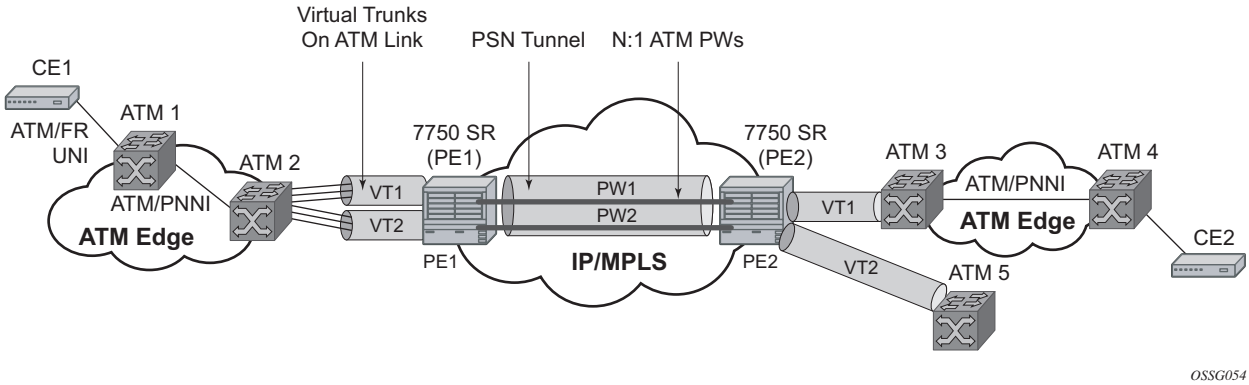


Figure 2: VT Application Example

The virtual trunk (VT) SAP on a PE is identified by a tuple (port, VPI-range) meaning that all cells arriving on the specified port within the specified VPI range are fed into a single ATM pseudowire for transport across the IP/MPLS network. Note that a user can configure the whole ATM port as a

VT and does not need to specify a VPI range. No VPI/VCI translation is performed on ingress or egress. Cell order is maintained within a VT. Note that, as a special case, the two ATM ports could be on the same PE node.

By carrying all cells from all VPIs making up the VT in one pseudowire, a solution is provided that is both robust, for example no black holes on some VPIs but not others, as well as operationally efficient since the entire VT can be managed as a single entity from the Network Manager (single point for configuration, status, alarms, statistics, etc.).

ATM virtual trunks use PWE3 N:1 ATM cell mode encapsulation to provide a cell-mode transport, supporting all AAL types, over the MPLS network. Cell concatenation on a pseudowire packet is supported. The SDP can be of an MPLS or a GRE type.

The ATM pseudowire is initiated using Targeted LDP (TLDP) signaling (defined in RFC 4447, *Pseudowire Setup and Maintenance using LDP*). In this application, there is no ATM signaling on the gateway nodes since both endpoints of the MPLS network are configured by the network operator. ATM signaling between the ATM nodes is passed transparently over the VT (along with user traffic) from one ATM port on a PE to another ATM port on a remote (or the same) SR PE.

Traffic Management Support

Ingress Network Classification

Classification is based on the EXP value of the pseudowire label and EXP-to-FC mapping is determined by the network ingress QoS policy.

Ingress Queuing and Shaping on the IOM

Each SAP of an ATM VLL has an associated single ingress service queue on the IOM. The default QoS policy configures this queue to have CIR=0 and PIR=line rate. Other QoS policies can be applied if they specify a single service queue. Applying a non-default QoS policy allows the CIR/PIR of the incoming traffic to be controlled, regardless of whether ATM policing is configured, and provides queuing and shaping to smooth traffic flows on the ingress of the network.

Egress Queuing and Shaping on the IOM

Each SAP of an ATM VLL has a single associated egress service queue on the IOM. The default QoS policy configures this queue to have CIR=0 and PIR=line rate. Other QoS policies can be applied if they specify a single service queue. Applying a non-default QoS policy allows the CIR/PIR of the outgoing traffic to be controlled, regardless of whether ATM shaping is configured.

Egress Shaping/Scheduling

Each SAP of an ATM VLL has an egress ATM traffic descriptor associated with it. The default traffic descriptor has service category UBR with zero MIR, resulting in endpoints associated with this descriptor being scheduled at the lowest priority on the ATM MDA. Egress traffic may be shaped or scheduled, depending on the configuration of the egress ATM traffic descriptor associated with the SAP. Table 2 provides details of how the different service categories and shaping settings affect egress transmission rates.

Shaping applies to CBR, rtVBR and nrtVBR service categories and results in cells being transmitted in such a way as to satisfy a downstream ATM UPC function. In particular, the transmission rate will be limited (in the case of CBR, there is a hard limit of PIR, while rtVBR/nrtVBR will transmit at SIR with short (constrained by MBS) bursts of up to PIR), and the inter-cell gap will also be controlled.

Service category UBR and rtVBR are scheduled on the WRR scheduler with the configured rates (MIR for UBR+) determining the weight applied to the flow. Weights are between 1 and 255 and are determined by a formula applied to the configured rate. UBR flows (for example, those with no MIR) receive a weight of 1 and the maximum weight of 255 is reached by flows with configured rates of around 8 Mbps. Scheduling does not apply a limit to the transmission rate; the available port bandwidth is shared out by the scheduler according to the weight, so if other flows are quiescent, a given flow may burst up to port bandwidth.

Shaping and scheduling of egress ATM VLL traffic is performed entirely at the ATM layer and is therefore not forwarding-class-aware. If the offered rate is greater than can be transmitted towards the customer (either because the shaping rate limits transmission or because the SAP does not receive sufficient servicing in the weighed round-robin used for scheduled SAPs), the per-VC queue will back up and this will trigger the congestion control mechanisms in the MDA queues or in the IOM service egress queues associated with the SAP. For AAL5 SDU VLLs, these discards occur at the AAL5 SDU level. For N-to-1 VLLs, these discards occur at the level of the cell or a block of cells when cell concatenation is enabled.

Table 2: Behavior and Relative Priorities

Flow type	Transmission rate	Priority
shaped CBR	Limited to configured PIR	Strict priority over all other traffic
shaped rtVBR	Limited to configured SIR, but with bursts up to PIR within MBS	Strict priority over all but shaped CBR
shaped nrtVBR	Limited to configured SIR, but with bursts up to PIR within MBS	Strict priority over all scheduled traffic

Table 2: Behavior and Relative Priorities (Continued)

Flow type	Transmission rate	Priority
scheduled nrtVBR	Weighted share (according to SIR) of port bandwidth remaining after shaped traffic has been exhausted	In the same WRR scheduler as UBR+ and UBR
scheduled UBR+	Weighted share (according to MIR) of port bandwidth remaining after shaped traffic has been exhausted	In the same WRR scheduler as nrtVBR and UBR
scheduled UBR	Weighted share (with weight of 1) of port bandwidth remaining after shaped traffic has been exhausted	In the same WRR scheduler as nrtVBR and UBR+

Circuit Emulation Services (Cpipe)

Mobile Infrastructure

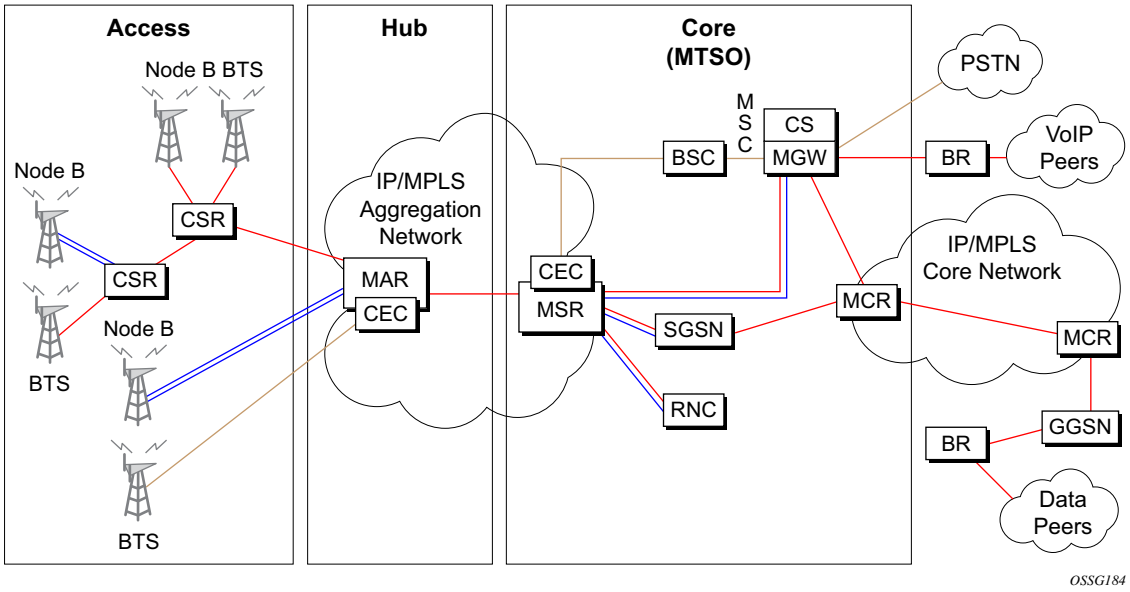


Figure 3: Mobile Infrastructure

Table 3: Mobile Infrastructure Definitions

Cellsite Backhaul Type	CSR Role	Transport Acronyms
Microwave	Circuit emulation	CSR: Cellsite Service Router
xDSL	ATM IMA termination into pseudowire	MAR: Mobile Aggregation Router
Fiber, dark or light	Ethernet VLL switching	MSR: Mobile Service Router
ATM, ATM IMA	IP/MPLS aggregation	CEC: Circuit Emulation Concentrator
Leased line		MCR: Mobile Core Router
		BR: Border Router

Packet infrastructure is required within 2G, 2.5G and 3G mobile networks to handle SMS messaging, web browsing and emerging applications such as streaming video, gaming and video

on demand. Within existing 2.5G and 3G mobile networks, ATM is defined as the transport protocol. Within existing 2G networks, TDM is defined as the transport protocol. Due to the relatively low bit rate of existing handsets, most cell sites use 2-10 DS1s or E1s to transport traffic. When using ATM over multiple DS1/E1 links, Inverse Multiplexing over ATM (IMA) is very effective for aggregating the available bandwidth for maximum statistical gain and providing automatic resilience in the case of a link failure. Also, multiple DS1s or E1s are required to transport the 2G voice traffic.

Typically, low cost devices are used at the many cell sites to transport multiple DS1 or E1 using ATM/IMA and TDM over an Ethernet/MPLS infrastructure. In Alcatel-Lucent applications, the circuit emulation would currently be performed using the 7705 SAR. This could be performed by DMXplore at the cell site. However, a large number of cell sites aggregate into a single switching center. Book-ending 7705 SAR nodes would require a very large number of systems at the switching center (Figure 3). Therefore, a channelized OC3/STM1 solution is much more efficient at the switching center. With the introduction of a channelized OC3/STM1 CES CMA/MDA in the 7750 SR, Alcatel-Lucent can provide a converged, flexible solution for IP/MPLS infrastructures for 2G/2.5G/3G mobile networks supporting both the CES (by CES CMA/MDA) and ATM/IMA transported traffic (by the ASAP MDA).

Circuit Emulation Modes

Two modes of circuit emulation are supported, unstructured and structured. Unstructured mode is supported for DS1 and E1 channels per RFC 4553, *Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)*. Structured mode is supported for n*64 kbps circuits as per RFC 5086, *Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)*. In addition, DS1, E1 and n*64 kbps circuits are supported (per MEF8). TDM circuits are optionally encapsulated in MPLS or Ethernet as per the referenced standards in the following figures.

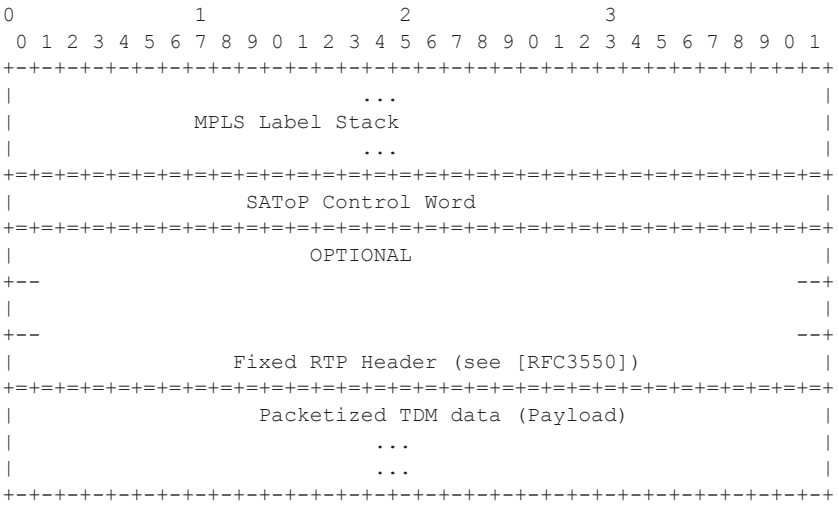


Figure 4: RFC 4553 (SAToP) MPLS PSN Encapsulation

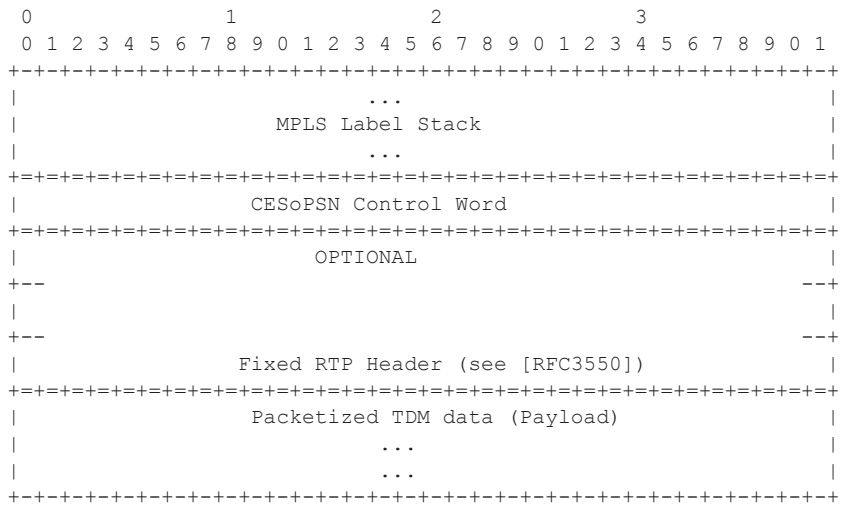


Figure 5: CESoPSN Packet Format for an MPLS PSN

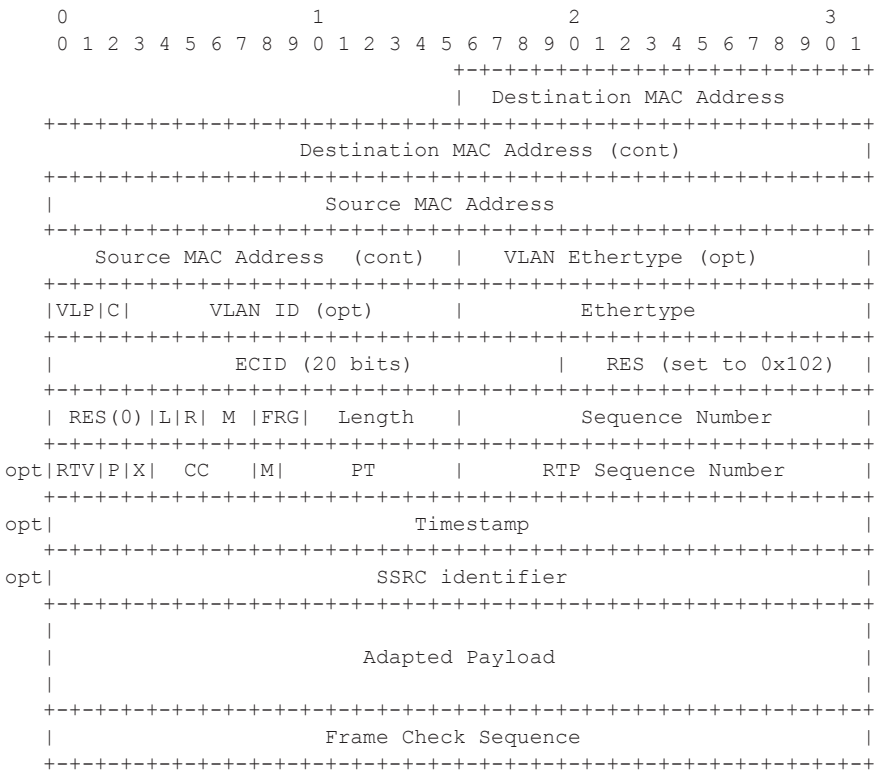


Figure 6: MEF8 PSN Encapsulation

Circuit Emulation Parameters

Circuit Emulation Modes

All channels on the CES CMA/MDA are supported as circuits to be emulated across the packet network. Structure aware mode is supported for n*64 kbps channel groups in DS1 and E1 carriers. Fragmentation is not supported for circuit emulation packets (structured or unstructured).

For DS1 and E1 unstructured circuits, the framing can be set to unframed. When channel group 1 is created on an unframed DS1 or E1, it is automatically configured to contain all 24 or 32 channels respectively.

N*64 kbps circuit emulation supports basic and Channel Associated Signaling (CAS) options for timeslots 1-31 (channels 2-32) on E1 carriers and channels 1-24 on DS1 carriers. CAS in-band is supported, therefore no separate pseudowire support for CAS is provided. CAS option can be enabled or disabled for all channel groups on a given DS1 or E1. If CAS operation is enabled, timeslot 16 (channel 17) cannot be included in the channel group on E1 carriers. CCS operation is not supported.

Absolute Mode Option

For all circuit emulation channels except those with differential clock sources, RTP headers in absolute mode can be optionally enabled (off by default). For circuit emulation channels which use differential clock sources, this configuration is blocked. All channel groups on a given DS1 or E1 can be configured for the same mode of operation.

When enabled for absolute mode operation, an RTP header will be inserted. On transmit, the CES IWF will insert an incrementing (by 1 for each packet) timestamp into the packets. All other fields will be set to zero. The RTP header will be ignored on receipt. This mode is enabled for interoperability purposes only for devices which require an RTP header to be present.

Payload Size

For DS3, E3, DS1 and E1 circuit emulation, the payload size can be configurable in number of octets. The default values for this parameter are shown in Table 4. Unstructured payload sizes can be set to a multiple of 32 octets and minimally be 64 octets.

Table 4: Unstructured Payload Defaults

TDM Circuit	Default Payload Size
DS1	192 octets
E1	256 octets

For n*64 kbps circuits, the number of octets or DS1/E1 frames to be included in the TDM payload needs to be configurable in the range 4 to128 DS1/E1 frames in increments of 1 or the payload size in octets. The default number of frames is shown in the table below with associated packet sizes. For the number of 64 kbps channels included (N), the following number of frames defaults apply for no CAS: N=1, 64 frames; 2<=N<= 4, 32 frames; 5<=N<= 15, 16 frames; N>=16, 8 frames.For CAS circuits, the number of frames can be 24 for DS1 and 16 for E1 which yields a payload size of N*24 octets for T1 and N*16 octets for E1. For CAS, the signaling portion is an additional ((N+1)/2) bytes where N is the number of channels. The additional signaling bytes are not included in the TDM payload size, although they are included in the actual packet size shown in Table 5.

The full ABCD signaling value can be derived before the packet is sent. This occurs for every 24 frames for DS1 ESF and every 16 frames for E1. Note that for DS1 SF, ABAB signaling is actually sent as SF framing only supports AB signaling every 12 frames.

Table 5: Structured Number of Frames Defaults

Num Timeslots	no CAS			DS1 CAS		E1 CAS	
	num-frames default	Default Payload	Minimum Payload	Pay-load (24 frames)	Packet Size	Pay-load (16 frames)	Packet Size
1	64	64	40	24	25	16	17
2	32	64	64	48	49	32	33
3	32	96	96	72	74	48	50
4	32	128	128	96	98	64	66
5	16	80	80	120	123	80	83
6	16	96	96	144	147	96	99
7	16	112	112	168	172	112	116
8	16	128	128	192	196	128	132
9	16	144	144	216	221	144	149
10	16	160	160	240	245	160	165

Table 5: Structured Number of Frames Defaults (Continued)

Num Timeslots	no CAS			DS1 CAS		E1 CAS	
	num-frames default	Default Payload	Minimum Payload	Pay-load (24 frames)	Packet Size	Pay-load (16 frames)	Packet Size
11	16	176	176	264	270	176	182
12	16	192	192	288	294	192	198
13	16	208	208	312	319	208	215
14	16	224	224	336	343	224	231
15	16	240	240	360	368	240	248
16	8	128	128	384	392	256	264
17	8	136	136	408	417	272	281
18	8	144	144	432	441	288	297
19	8	152	152	456	466	304	314
20	8	160	160	480	490	320	330
21	8	168	168	504	515	336	347
22	8	176	176	528	539	352	363
23	8	184	184	552	564	368	380
24	8	192	192	576	588	384	396
25	8	200	200	NA	NA	400	413
26	8	208	208	NA	NA	416	429
27	8	216	216	NA	NA	432	446
28	8	224	224	NA	NA	448	462
29	8	232	232	NA	NA	464	479
30	8	240	240	NA	NA	480	495
31	8	248	248	NA	NA	NA	NA

NOTE: num-frames DS1 CAS are multiples of 24; num-frames E1 is a multiple of 16.

Jitter Buffer

For each circuit, the maximum receive jitter buffer are configurable. Playout from this buffer starts when the buffer is 50% full to give an operational packet delay variance (PDV) equal to 75% of the maximum buffer size. The default value for the jitter buffer is nominally 5 ms. However, for lower speed N*64kbps circuits and CAS circuits, the following default values are used to align with the default number of frames (and resulting packetization delay) to allow at least two frames to be received before starting to playout the buffer. The jitter buffer is at least four times the packetization delay. The following default jitter buffer values for structured circuits apply:

Basic CES (DS1 & E1):

N=1, 32 ms

2<=N<= 4, 16 ms

5<=N<=15, 8 ms

N>=16, 5 ms

CES Circuit Operation

The circuit status can be tracked to be either up, loss of packets or administratively down. Statistics are available for the number of in service seconds and the number of out of service seconds when the circuit is administratively up.

Jitter buffer overrun and underrun counters are available by statistics and optionally logged while the circuit is up. On overruns, excess packets are discarded and counted. On underruns, all ones are sent for unstructured circuits. For structured circuits, all ones or a user defined data pattern is sent based on configuration. Also, if CAS is enabled, all ones or a user defined signaling pattern is sent based on configuration.

For each CES circuit, alarms can be optionally disabled/enabled for stray packets, malformed packets, packet loss, receive buffer overrun and remote packet loss. An alarm is raised if the defect persists for 3 seconds, and cleared when defect no longer persists for 10 seconds. These alarms are logged and trapped when enabled.

Services for Transporting CES Circuits

Each circuit can be optionally encapsulated in MPLS, Ethernet packets. Circuits encapsulated in MPLS use circuit pipes (Cpipes) to connect to the far-end circuit. Cpipes support either SAP-spoke SDP or SAP-SAP connections. Cpipes are supported over MPLS and GRE tunnels. Cpipe's default service MTU is set to 1514 bytes.

Circuits encapsulated in Ethernet can be selected as a SAP in Epipes. Circuits encapsulated in Ethernet can be SAP-spoke SDP connections or Ethernet CEM SAP to Ethernet SAP for all valid epipe SAPs. Circuits requiring CEM SAP — CEM SAP connections use Cpipes. A local and remote EC-ID and far-end destination MAC address can be configurable for each circuit. The CMA/MDA's MAC address will be used as the source MAC address for these circuits.

For all service types, there are deterministic PIR=CIR values with class=EF parameters based on the circuit emulation parameters.

All circuit emulation services support the display of status of up, loss of packets (LOP) or admin down. Also, any jitter buffer overruns or underruns are logged.

Non-stop services are supported for Cpipes and CES over Epipes.

Network Synchronization Considerations

Each OC-3/STM-1 port can be independently configured to be loop-timed or node-timed. Each OC-3/STM-1 port can be configured to be a timing source for the node.

Each DS-1 or E-1 channel without CAS signaling enabled can be independently configured to be loop-timed, node-timed, adaptive-timed or differential-timed. Each DS-1 or E-1 channel with CAS signaling enabled can be independently configured to be loop-timed or node-timed. Adaptive-timed and differential-timed are not supported on DS-1 or E-1 channels with CAS signaling enabled.

A CES circuit's adaptive recovered clock can be used a timing reference source for the node (ref1 or ref2). This is required to distribute network timing to network elements which only have packet connectivity to the network. One timing source on the CMA/MDA can be monitored for timing integrity. Both timing sources can be monitored if they are configured on separate CMA/MDAs while respecting the timing subsystem slot requirements. If a CES circuit is being used for adaptive clock recovery at the remote end (such that the local end is now an adaptive clock master), it is recommended to set the DS-1/E-1 to be node-timed to prevent potential jitter issues in the recovered adaptive clock at the remote device.

For differential-timed circuits, the following timestamp frequencies are supported: 103.68 MHz (for recommended >100MHz operation), 77.76 MHz (for interoperability with SONET/SDN based systems such as TSS-5) and 19.44 MHz (for Y.1413 compliance).

Adaptive and differential timing recovery must comply with published jitter and wander specifications (G.823, G.824 and G.8261) for traffic interfaces under typical network conditions and for synchronous interfaces under specified packet network delay, loss and delay variance (jitter) conditions. The packet network requirements to meet the synchronous interface requirements are to be determined during the testing phase.

On the 7450 ESS and 7750 SR CES CMA, a BITS port is also provided. The BITS port can be used as one of the two timing reference sources in the system timing subsystem. The operation of BITS ports configured as ref1 or ref2 is the same as existing ports configured as ref1 and ref2 with all options supported. The operation of the 7450 ESS or 7750 SR BITS source is unchanged and the BITS ports are not available on the CES MDAs (only SF/CPM BITS are currently available).

Cpipe Payload

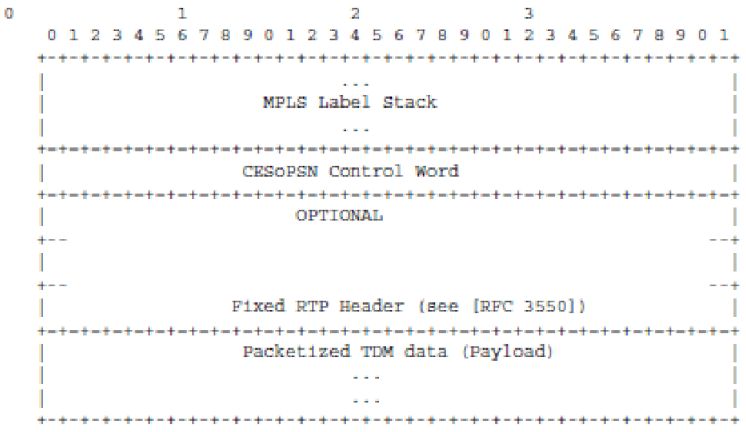


Figure 7: CESoPSN MPLS Encapsulation

Figure 7 shows the format of the CESoPSN TDM payload (with and without CAS) for packets carrying trunk-specific 64 kb/s service. In CESoPSN, the payload size is dependent on the number of timeslots used.

Ethernet Pipe (Epipe) Services

This section provides information about the Epipe service and implementation notes.

Topics in this section include:

Epipe Service Overview

An Epipe service is Alcatel-Lucent's implementations of an Ethernet VLL based on the IETF "Martini Drafts" (draft-martini-l2circuit-trans-mpls-08.txt and draft-martini-l2circuit-encapmpls-04.txt) and the IETF Ethernet Pseudo-wire Draft (draft-so-pwe3-ethernet-00.txt).

An Epipe service is a Layer 2 point-to-point service where the customer data is encapsulated and transported across a service provider's IP, MPLS or PBB VPLS network. An Epipe service is completely transparent to the subscriber's data and protocols. The , 7750 SR, and Epipe service does not perform any MAC learning. A local Epipe service consists of two SAPs on the same node, whereas a distributed Epipe service consists of two SAPs on different nodes. SDPs are not used in local Epipe services.

Each SAP configuration includes a specific port/channel on which service traffic enters the , 7750 SR, or from the customer side (also called the access side). Each port is configured with an encapsulation type. If a port is configured with an IEEE 802.1Q (referred to as Dot1q) encapsulation, then a unique encapsulation value (ID) must be specified.

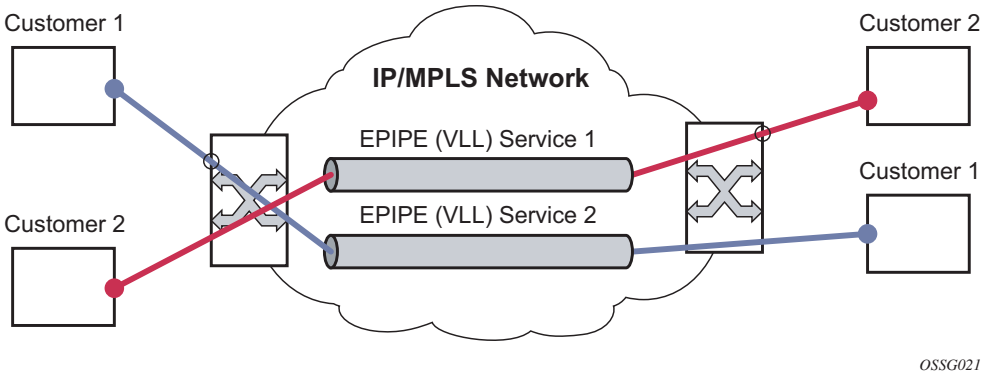


Figure 8: Epipe/VLL Service

Epipe Service Pseudowire VLAN Tag Processing

Distributed Epipe services are connected using a pseudowire, which can be provisioned statically or dynamically and is represented in the system as a spoke SDP. The spoke SDP can be configured to process zero, one or two VLAN tags as traffic is transmitted and received; see Table 6 and Table 7 for configuration details. In the transmit direction, VLAN tags are added to the frame being sent. In the received direction, VLAN tags are removed from the frame being received. This is analogous to the SAP operations on a null, dot1q and QinQ SAP.

The system expects a symmetrical configuration with its peer; specifically, it expects to remove the same number of VLAN tags from received traffic as it adds to transmitted traffic. When removing VLAN tags from a spoke SDP, the system attempts to remove the configured number of VLAN tags (see below for configuration details). If fewer tags are found, the system removes the VLAN tags found and forwards the resulting packet. As some of the related configuration parameters are local and not communicated in the signaling plane, an asymmetrical behavior cannot always be detected and so cannot be blocked. With an asymmetrical behavior, protocol extractions will not necessarily function as they would with a symmetrical configurations, thus resulting in an unexpected operation.

The VLAN tag processing is configured as follows on a spoke SDP in an Epipe service:

- Zero VLAN tags processed—This requires the configuration of **vc-type ether** under the spoke SDP, or in the related **pw-template**.
- One VLAN tag processed—This requires one of the following configurations:
  - → **vc-type vlan** under the spoke SDP or in the related **pw-template**
  - → **vc-type ether** and **force-vlan-vc-forwarding** under the spoke SDP or in the related **pw-template**
- Two VLAN tags processed—This requires the configuration of **force-qinq-vc-forwarding** under the spoke SDP or in the related **pw-template**.

The **pw-template** configuration provides support for BGP VPWS services.

The following restrictions apply to VLAN tag processing:

- The configuration of **vc-type vlan** and **force-vlan-vc-forwarding** is mutually exclusive.
- **force-qinq-vc-forwarding** can be configured with the spoke SDP signaled as either **vc-type ether** or **vc-type vlan**.
- The following are not supported with **force-qinq-vc-forwarding** configured under the spoke SDP, or in the related **pw-template**:
  - → Multi-segment pseudowires.
  - → BGP VPWS routes are not accepted over an iBGP session.

→ ETH-CFM MIPs and MEPs are not supported on dynamically signaled BGP QinQ PWs.

Table 6 and Table 7 describe the VLAN tag processing with respect to the zero, one and two VLAN tag configuration described above for the VLAN identifiers, Ether type, ingress QoS classification (dot1p/DE) and QoS propagation to the egress (which can be used for egress classification and/or to set the QoS information in the innermost egress VLAN tag).

Table 6: Epipe Spoke SDP VLAN Tag Processing: Ingress

Ingress (received on spoke SDP)	Zero VLAN tags	One VLAN tag	Two VLAN tags
VLAN identifiers	N/A	Ignored	Both inner and outer ignored
Ether type (to determine the presence of a VLAN tag)	N/A	0x8100 or value configured under **sdp vlan-vc-etype**	Both inner and outer VLAN tags: 0x8100, or outer VLAN tag value configured under **sdp vlan-vc-etype** (inner VLAN tag value must be 0x8100)
Ingress QoS (dot1p/DE) classification	N/A	Ignored	Both inner and outer ignored
QoeE (dot1p/DE) propagation to egress	Dot1p/DE= 0	Dot1p/DE taken from received VLAN tag	Dot1p/DE taken from inner received VLAN tag

Table 7: Epipe Spoke SDP VLAN Tag Processing: Egress

Egress (sent on mesh or spoke SDP)	Zero VLAN tags	One VLAN tag	Two VLAN tags
VLAN identifiers (set in VLAN tags)	N/A	• the **vlan-vc-tag** value configured in **pw-template** or under the spoke SDP or • taken from the inner tag received on a QinQ SAP or QinQ spoke SDP or • taken from the VLAN tag received on a dot1q SAP or spoke SDP (with **vc-type vlan** or **force-vlan-vc-forwarding**) or • taken from the outer tag received on a qtag.* SAP or • 0 if there is no service delimiting VLAN tag at the ingress SAP or spoke SDP	Both inner and outer VLAN tag: • the **vlan-vc-tag** value configured in **pw-template** or under the spoke SDP or • taken from the inner tag received on a QinQ SAP or QinQ spoke SDP or • taken from the VLAN tag received on a dot1q SAP or spoke SDP (with **vc-type vlan** or **force-vlan-vc-forwarding**) or • taken from the outer tag received on a qtag.* SAP or • 0 if there is no service delimiting VLAN tag at the ingress SAP or spoke SDP

Table 7: Epipe Spoke SDP VLAN Tag Processing: Egress (Continued)

Egress (sent on mesh or spoke SDP)	Zero VLAN tags	One VLAN tag	Two VLAN tags
Ether type (set in VLAN tags)	N/A	0x8100 or value configured under **sdp vlan-vc-etype**	Both inner and outer VLAN tags: 0x8100, or outer VLAN tag value configured under **sdp vlan-vc-etype** (inner VLAN tag value will be 0x8100)
Egress QoS (dot1p/DE) (set in VLAN tags)	N/A	Taken from the inner most ingress service delimiting tag: • the inner tag received on a QinQ SAP or QinQ spoke SDP or • taken from the VLAN tag received on a dot1q SAP or spoke SDP (with **vc-type vlan** or **force-vlan-vc-forwarding**) or • taken from the outer tag received on a qtag.* SAP or • 0 if there is no service delimiting VLAN tag at the ingress SAP or spoke SDP. Note that neither the inner nor outer dot1p/DE values can be explicitly set.	Both inner and outer dot1p/DE: Taken from the innermost ingress service delimiting tag: • the inner tag received on a QinQ SAP or QinQ spoke SDP or • taken from the VLAN tag received on a dot1q SAP or spoke SDP (with **vc-type vlan** or **force-vlan-vc-forwarding**) or • taken from the outer tag received on a qtag.* SAP or • 0 if there is no service delimiting VLAN tag at the ingress SAP or spoke SDP. Note that neither the inner nor outer dot1p/DE values can be explicitly set.

Any non-service delimiting VLAN tags are forwarded transparently through the Epipe service. SAP egress classification is possible on the outer most customer VLAN tag received on a spoke SDP using the **ethernet-ctag** parameter in the associated SAP egress QoS policy.

Epipe Up Operational State Configuration Option

By default, the operational state of the Epipe is tied to the state of the two connections that comprise the Epipe. If either of the connections in the Epipe are operationally down, the Epipe service that contains that connection will also be operationally down. The operator does have the ability to configure a single SAP within an Epipe not to affect the operational state of that Epipe using the optional command **ignore-oper-state**. Within an Epipe, if a SAP that includes this optional command becomes operationally down state, the operational state of the Epipe will not transition to down. The operational state of the Epipe will remain up. This does not change the fact that the SAP is down and no traffic will transit an operationally down SAP. Removing and adding this command on the fly will evaluate the service's operational state based on the SAPs and the addition or deletion of this command.

Service OAM (SOAM) designers may consider using this command if an UP MEP configured on the operationally down SAP within an Epipe is required to receive and process SOAM PDUs. When a service is operationally down, this is not possible. For SOAM PDUs to continue to arrive on an UP, MEP configured on the failed SAP the service must be operationally up. Consider the case where an UP MEP is placed on a UNI-N or E-NNI and the UNI-C on E-NNI peer is shutdown in such a way that it causes the SAP to enter an operational state Down.

Two connections must be configured within the Epipe, otherwise, the service will be operationally down regardless of this command. The **ignore-oper-state** functionality will only operate as intended when the Epipe has one ingress and one egress. This command is not to be used for Epipe services with redundant connections that provide alternate forwarding in case of failure, even though the CLI does not prevent this configuration.

Support is available on Ethernet SAPs configured on ports or Ethernet SAPs configured on LAG. However, it is not allowed on SAPs using LAG profiles or if the SAP is configured on a LAG which has no ports.

Epipe with PBB

A pbb-tunnel may be linked to an Epipe to a B-VPLS. MAC switching and learning is not required for the point-to-point service (all packets ingressing the SAP are PBB encapsulated and forwarded to the PBB tunnel to the backbone destination MAC address and all the packets ingressing the B-VPLS destined for the ISID are PBB de-encapsulated and forwarded to the Epipe SAP. A fully specified backbone destination address must be provisioned for each PBB Epipe instance to be used for each incoming frame on the related I-SAP. If the backbone destination address is not found in the B-VPLS FDB then packets may be flooded through the B-VPLSs

All B-VPLS constructs may be used including B-VPLS resiliency and OAM. Not all generic Epipe commands are applicable when using a PBB tunnel.

Epipe over L2TPv3

The L2TPv3 feature provides a framework to transport Ethernet pseudowire services over an IPv6-only network without MPLS. This architecture relies on the abundance of address space in the IPv6 protocol to provide unique far-end and local-end addressing that uniquely identify each tunnel and service binding.

L2TPv3 provides the capability of transporting multiple EPipes (up to 16K per system), by binding multiple IPv6 addresses to each node and configuring one SDP per Epipe.

As the IPv6 addressing uniqueness identifies the customer and service binding, the L2TPv3 control plane is disabled in this mode.

L2TPv3 is supported on non-12e 7750 SR and (mixed mode) and platforms, in mode D with FP2+ (FP3 recommended).

ETH-CFM is supported for OAM services.

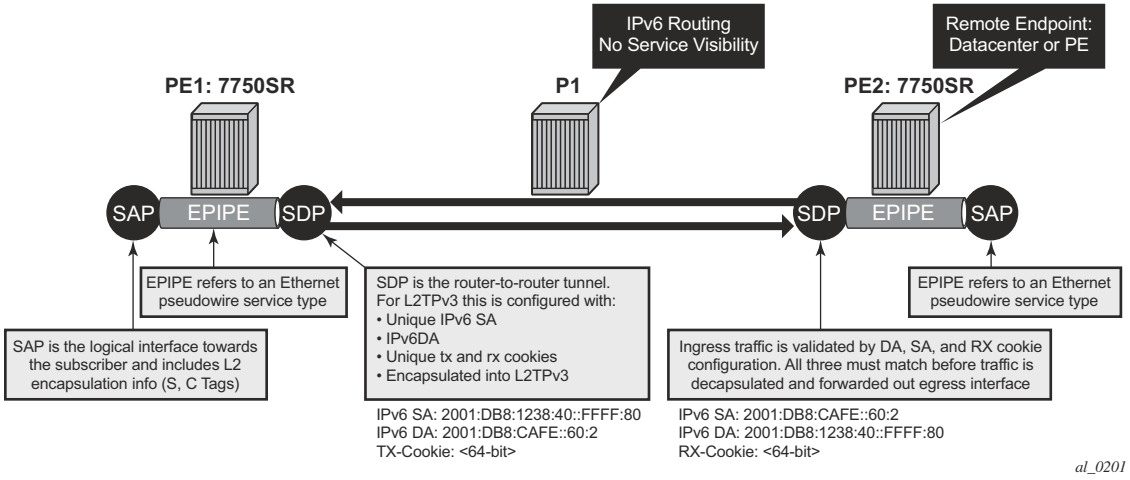


Figure 9: L2TPv3 SDP Illustration

Ethernet Interworking VLL

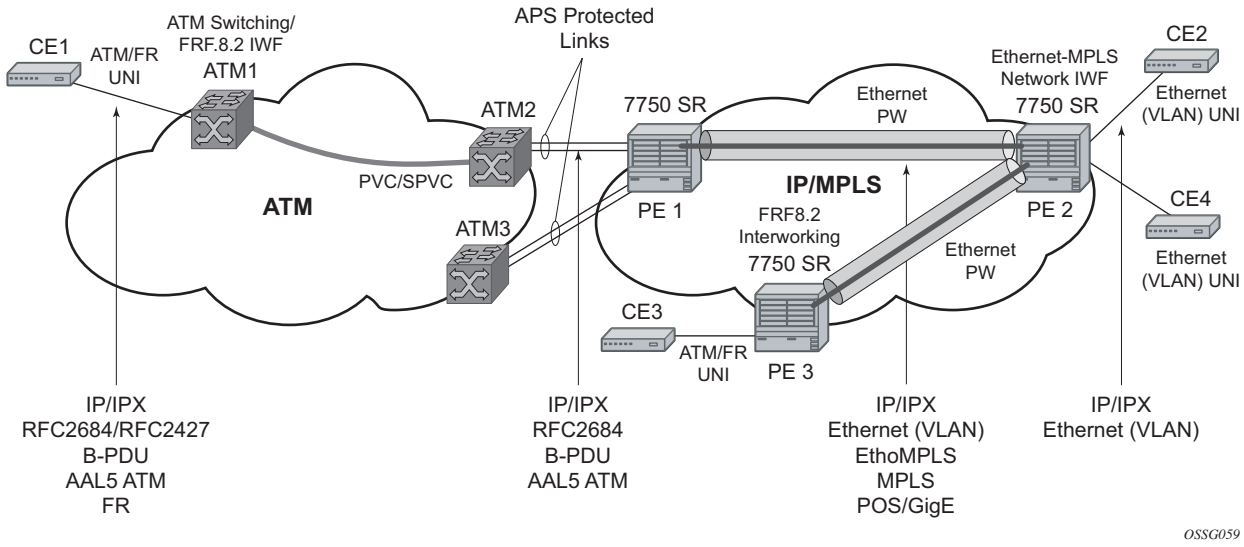


Figure 10: Application of Ethernet Interworking VLL Example

Figure 10 provides an example of an Ethernet interworking VLL. The Ethernet interworking VLL provides a point-to-point Ethernet VLL service between Frame-Relay-attached users, ATM attached users, and Ethernet-attached users across an IP/MPLS packet switched network. It effectively provides ATM and FR bridged encapsulation termination on the existing Epipe service of the 7750 SR.

The following connectivity scenarios are supported:

- A Frame Relay or ATM user connected to a ATM network communicating with a Ethernet user connected to a 7750 SR PE node on a IP/MPLS network.
- A Frame Relay or ATM user connected to 7750 SR PE node communicating with an Ethernet user connected to a 7750 SR PE node on a IP/MPLS network. This feature supports local cross-connecting when these users are attached to the same 7750 SR PE node.

Users attach over an ATM UNI with RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*, tagged/untagged bridged Ethernet PDUs, a FR UNI using RFC 2427, *Multiprotocol Interconnect over Frame Relay*, tagged/untagged bridged Ethernet PDUs, or an Ethernet tagged/untagged UNI interface. However, the VCI/VPI and the data-link control layer (DLCI) are the identifiers of the SAP in the case of ATM and FR respectively and the received tags are transparent to the service and are thus preserved.

The Ethernet pseudowire is established using Targeted LDP (TLDP) signaling and can use the **ether** or **vlan** VC types on the SDP. The SDP can be either an MPLS or GRE type.

VLL CAC

This feature provides a method to administratively account for the bandwidth used by VLL services inside an SDP which consists of RSVP LSPs.

The service manager keeps track of the available bandwidth for each SDP. The SDP available bandwidth is applied through a configured booking factor. An administrative bandwidth value is assigned to the spoke SDP. When a VLL service is bound to an SDP, the amount of bandwidth is subtracted from the adjusted available SDP bandwidth. When the VLL service binding is deleted from the SDP, the amount of bandwidth is added back into the adjusted SDP available bandwidth. If the total adjusted SDP available bandwidth is overbooked when adding a VLL service, a warning is issued and the binding is rejected.

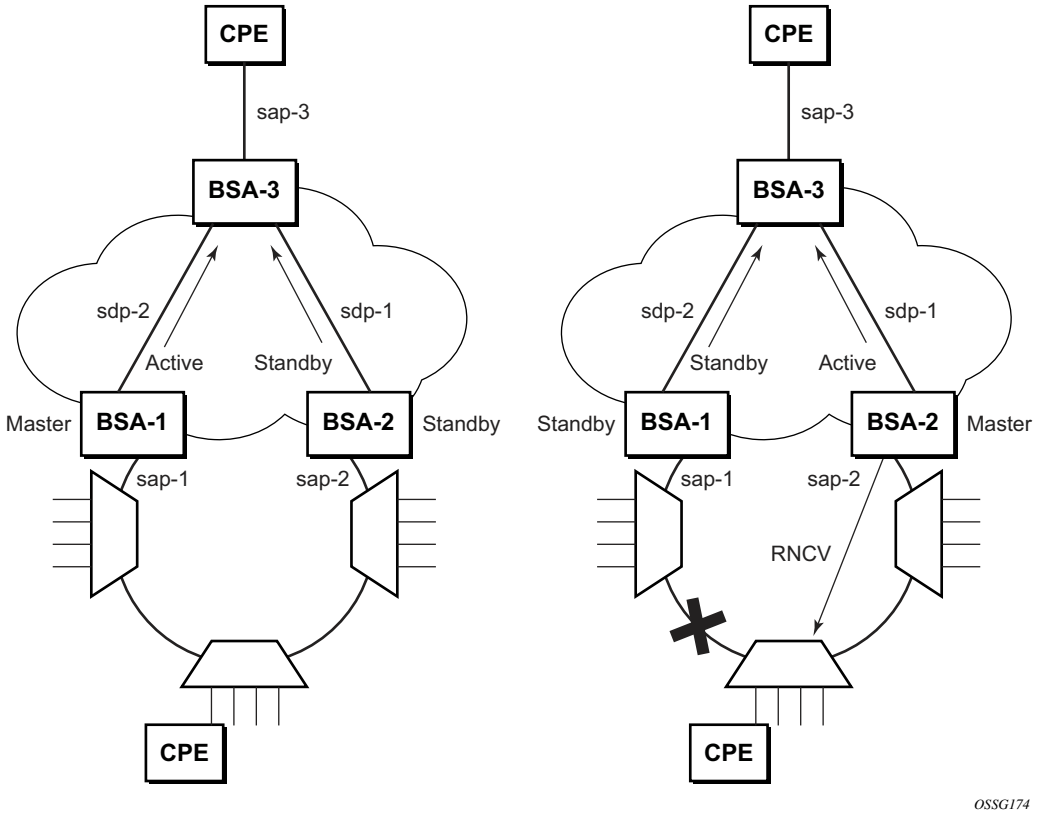
This feature does not guarantee bandwidth to a VLL service as there is no change to the datapath to enforce the bandwidth of an SDP by means such as shaping or policing of constituent RSVP LSPs.

MC-Ring and VLL

To support redundant VLL access in ring configurations, the multi-chassis ring feature is applicable to VLL SAPs. A conceptual drawing of the operation is shown in Figure 11. The given CPE which is connected behind the ring node has access to both BSA through the same VLAN provisioned in all ring nodes. There are two SAPs (with the same VLAN) provisioned on both nodes.

If a closed ring status occurs, one of the BSAs becomes the master and it will signal an active status bit on the corresponding VLL pseudowire. Similarly, the standby BSA will signal a standby status. With this information, the remote node can choose the correct path to reach the CPE. In case of a broken ring, the node that can reach the ring node that the given CPE is connected to by RNCV check, will become master and will signal corresponding status on its pseudowire.

The mapping of individual SAPs to the ring nodes is done statically through CLI provisioning. In order to keep the convergence time to a minimum, MAC learning must be disabled on the ring node so all CPE originated traffic is sent in both directions. If the status is oper-down on the SAP on the standby BSA, that part of the traffic will be blocked and not forwarded to the remote site.



MC-Ring in a Combination with VLL Service

For further information about Multi-Chassis Ring Layer 2 (with ESM), refer to the Advanced Configuration Guide.

Frame Relay VLL (Fpipe) Services

This section provides information about the Fpipe service and implementation notes.

Topics in this section include:

Frame Relay VLL

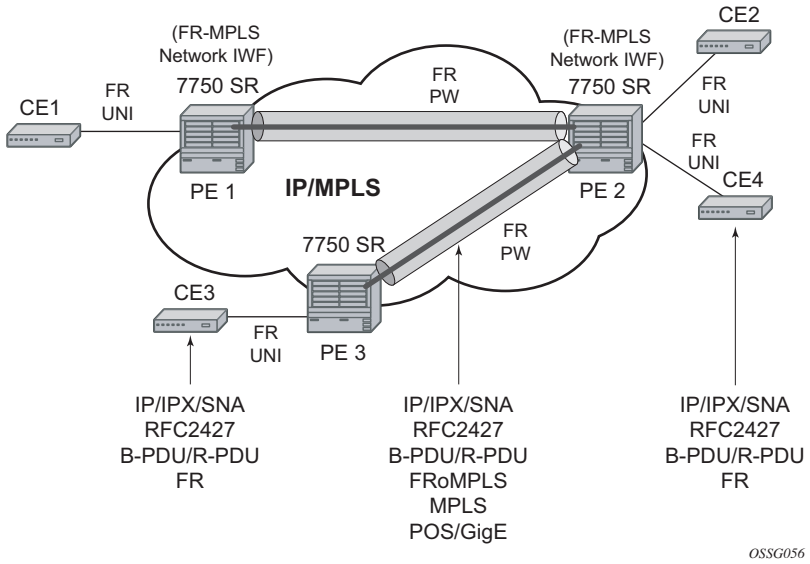


Figure 12: Application of a Frame Relay VLL Example

Figure 12 depicts an application of a Frame Relay VLL. The Frame Relay VLL (Fpipe) provides a point-to-point Frame Relay service between users connected to 7750 SR nodes on the IP/MPLS network. Users are connected to the 7750 SR PE nodes using Frame Relay PVCs. PE1, PE2, and PE3 receive a standard Q.922 Core frame on the Frame Relay SAP and encapsulate it into a pseudowire packet according to the 1-to-1 Frame Relay encapsulation mode in RFC 4619, *Encapsulation Methods for Transport of Frame Relay Over MPLS Networks*. The 7750 SR Frame Relay VLL feature supports local cross-connecting when the users are attached to the same 7750 SR PE node.

The FR pseudowire is initiated using Targeted LDP (TLDP) signaling as specified in RFC 4447, *Pseudowire Setup and Maintenance using LDP*. The SDP can be an MPLS or a GRE type.

Frame Relay-to-ATM Interworking (FRF.5) VLL

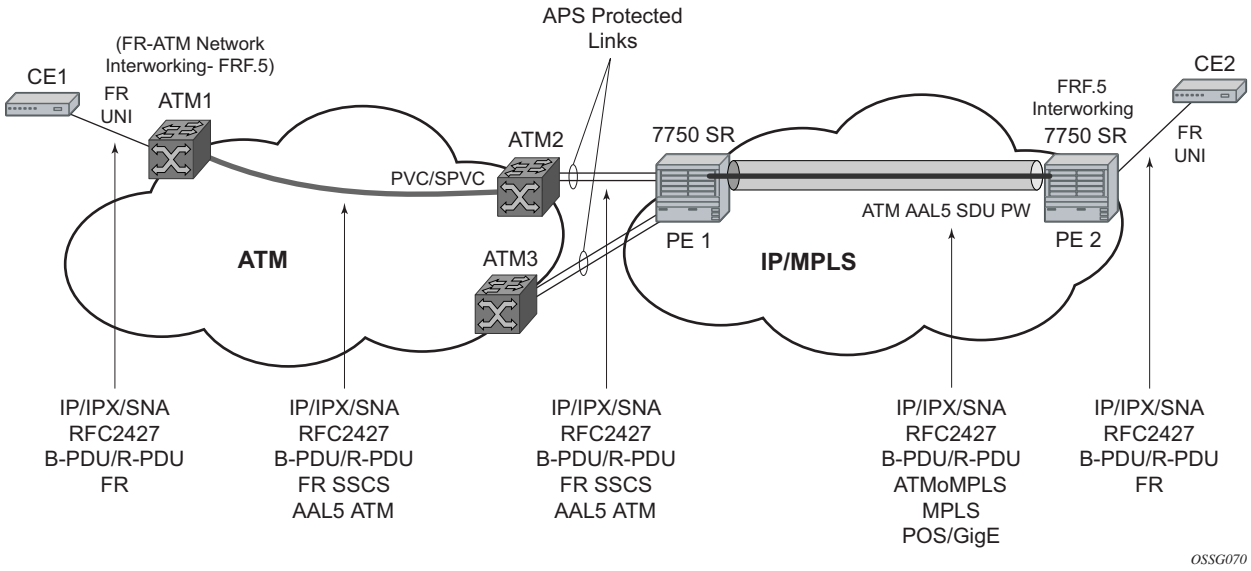


Figure 13: Frame Relay-to-ATM Network Interworking (FRF.5) VLL

Figure 13 provides an example of a point-to-point Frame Relay service between users where one user is connected to an existing ATM network, the other to a 7750 SR PE node on an IP/MPLS network.

This VLL is realized using an ATM AAL5 SDU pseudowire between the 7750 SR SR PE nodes. It is configured by adding a FR SAP to an Apipe service using vc-type atm-sdu. The 7750 SR SR PE2 node performs an FRF.5 interworking function to interwork the ingress and egress data paths in addition to the operations required in an FR and an ATM VLL.

The pseudowire is initiated using Targeted LDP signaling as specified in draft-ietf-pwe3-control-protocol-xx.txt. The SDP can be of an MPLS or a GRE type.

Traffic Management Support

Frame Relay Traffic Management

Traffic management of Frame Relay VLLs is achieved through the application of ingress and egress QoS policies to SAPs like other Frame Relay SAPs. No queuing occurs on the MDA; all queuing, policing and shaping occurs on the IOM and, as a result, traffic management is forwarding-class-aware. Forwarding classes may be determined by inspecting the DSCP marking of contained IP packets (for example) and this will determine both the queuing and the EXP bit setting of packets on a Frame Relay VLL.

Ingress SAP Classification and Marking

DE=0 frames are subject to the CIR marking algorithm in the IOM queue. Drop preference for these packets will follow the state of the CIR bucket associated with the ingress queue. The value is marked in the drop preference bit of the internal header and into the DE bit in the Q.922 frame header. DE=1 frames are classified into “out-of-profile” state and are not be overwritten by the CIR marking in the ingress IOM queue. The drop preference is set to high.

Egress Network EXP Marking

FC-to-EXP mapping is as per the Network Egress QoS policy. Marking of the EXP field in both label stacks is performed.

Ingress Network Classification

Classification is based on the EXP value of the pseudowire label and EXP-to-FC mapping is as per Network Ingress QoS policy.

IP Interworking VLL (Ipipe) Services

Ipipe VLL

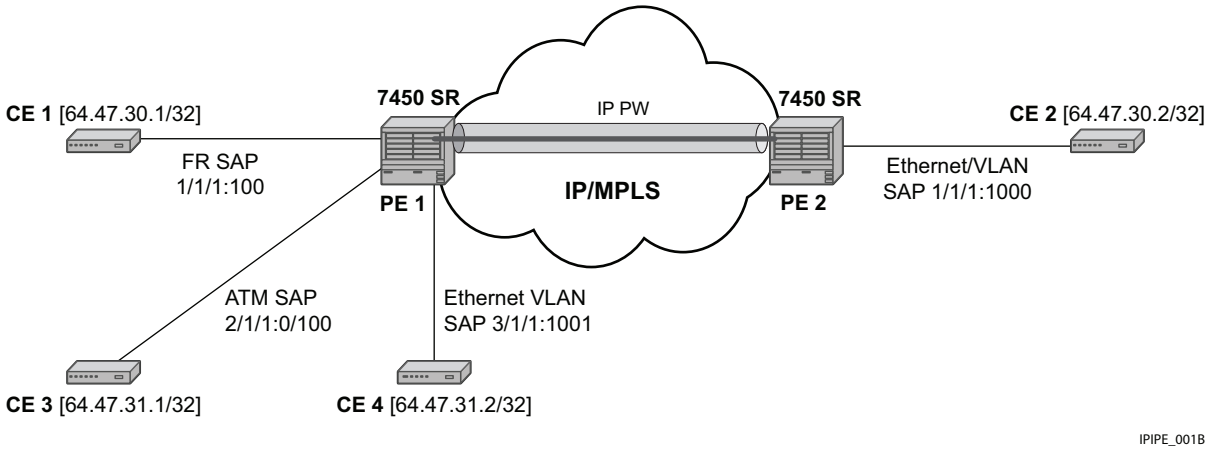


Figure 14: IP Interworking VLL (Ipipe)

Figure 14 provides an example of IP connectivity between a host attached to a point-to-point access circuit (FR, ATM, PPP) with routed PDU IPv4 encapsulation and a host attached to an Ethernet interface. Both hosts appear to be on the same LAN segment. This feature enables service interworking between different link layer technologies. A typical use of this application is in a Layer 2 VPN when upgrading a hub site to Ethernet while keeping the spoke sites with their existing Frame Relay or ATM IPv4 routed encapsulation.

The ATM SAP carries the IPv4 packet using RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*, *VC-Mux or LLC/SNAP routed PDU encapsulation.* The Frame Relay SAP

makes use of RFC 2427*, Multiprotocol Interconnect over Frame Relay,* routed PDU encapsulation of an IPv4 packet. A PPP interface makes use of RFC 1332, *The PPP Internet Protocol Control Protocol (IPCP)*, PPP IPCP encapsulation of an IPv4 packet. A Cisco HDLC SAP uses the routed IPv4 encapsulation. The pseudowire uses the IP Layer 2 transport pseudowire encapsulation type.

Note that the Ipipe is a point-to-point Layer 2 service. All packets received on one SAP of the Ipipe will be forwarded to the other SAP. No IP routing of customer packets occurs.

IP Interworking VLL Datapath

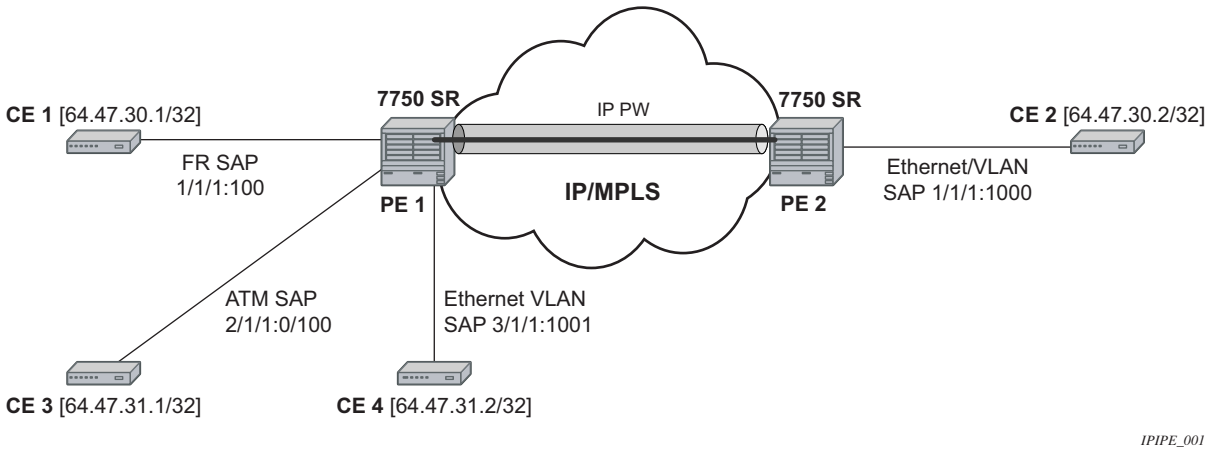


Figure 15: IP Interworking VLL Datapath

Figure 16Figure 15, PE 2 is manually configured with both CE 1 and CE 2 IP addresses. These are host addresses and are entered in /32 format. PE 2 maintains an ARP cache context for each IP interworking VLL. PE 2 responds to ARP request messages received on the Ethernet SAP. PE 2 responds with the Ethernet SAP configured MAC address as a proxy for any ARP request for CE 1 IP address. PE 2 silently discards any ARP request message received on the Ethernet SAP for an address other than that of CE 1. Likewise, PE 2 silently discards any ARP request message with the source IP address other than that of CE 2. In all cases, PE 2 keeps track of the association of IP to MAC addresses for ARP requests it receives over the Ethernet SAP.

In order to forward unicast frames destined to CE 2, PE 2 needs to know CE 2 MAC address. When the Ipipe SAP is first configured and administratively enabled, PE2 sends an ARP request message for CE 2 MAC address over the Ethernet SAP. Until an ARP reply is received from CE2, providing CE2's MAC address, unicast IP packets destined for CE2 will be discarded at PE2. IP broadcast and IP multicast packets are sent on the Ethernet SAP using the broadcast or direct-mapped multicast MAC address.

In order to forward unicast frames destined to CE 1, PE 2 validates the MAC destination address of the received Ethernet frame. It should match that of the Ethernet SAP. It then removes the Ethernet header and encapsulates the IP packet directly into a pseudowire without a control word. PE 1 removes the pseudowire encapsulation and forwards the IP packet over the Frame Relay SAP using RFC 2427, *Multiprotocol Interconnect over Frame Relay,* routed PDU encapsulation.

In order to forward unicast packets destined to CE1, PE2 validates the MAC destination address of the received Ethernet frame. If the IP packet is unicast, the MAC destination must match that of the Ethernet SAP. If the IP packet is multicast or broadcast, the MAC destination address must be an appropriate multicast or broadcast MAC address.

The other procedures are similar to the case of communication between CE 1 and CE 2, except that the ATM SAP and the Ethernet SAP are cross-connected locally and IP packets do not get sent over an SDP.

A PE does not flush the ARP cache unless the SAP goes administratively or operationally down. The PE with the Ethernet SAP sends unsolicited ARP requests to refresh the ARP cache every T seconds. ARP requests are staggered at an increasing rate if no reply is received to the first unsolicited ARP request. The value of T is configurable by user through the mac-refresh CLI command.

Extension to IP VLL for Discovery of Ethernet CE IP Address

VLL services provide IP connectivity between a host attached to a point to point access circuit (FR, ATM, PPP) with routed PDU encapsulation and a host attached to an Ethernet interface. Both hosts appear to be on the same IP interface. This feature is supported only for IPv4 payload.

In deployments where it is not practical for operators to obtain and configure their customer CE address, the following behaviors apply:

- A service comes up without prior configuration of the CE address parameter under both the SAP and the spoke SDP.
- Rely solely on received ARP messages from the Ethernet SAP attached CE device to update the ARP cache with no further check of the validity of the source IP address of the ARP request message and the IP address ARPed for.
- The LDP address list TLV to signal the learned CE IP address to the remote PE is supported. This is to allow the PE with the FR SAP to respond to an invFR ARP request message received from the FR attached CE device. Only Ethernet SAP and FR SAP can learn the CE address through ARP and invFR ARP respectively. The router does not support invATM ARP on an ATM interface.

VLL Ethernet SAP Procedures

The operator can enable the following CE address discovery procedures by configuring the **ce-address-discovery** in the **config>service>ipipe** context.

- The service is brought up without the CE address parameter configured at either the SAP or the spoke SDP.
- The operator cannot configure the **ce-address** parameter under the **config>service>ipipe>sap** or **config>service>ipipe>spoke-sdp** context when the **ce-address-discovery** in the **config>service>ipipe** context is enabled. Conversely, the operator is not allowed to enable the **ce-address-discovery** option under the Ipipe service if it has a SAP and/or spoke SDP with a user-entered **ce-address** parameter.

- While an ARP cache is empty, the PE does not forward unicast IP packets over the Ethernet SAP but forwards multicast/broadcast packets.
- The PE waits for an ARP request from the CE to learn both IP and MAC addresses of the CE. Both entries are added into the ARP cache. The PE accepts any ARP request message received over Ethernet SAP and updates the ARP cache IP and MAC entries with no further check of the source IP address of the ARP request message or of the IP address being ARPed.
- The , 7750 SR, and routers will always reply to a received ARP request message from the Ethernet SAP with the SAP MAC address and a source IP address of the IP address being ARPed without any further check of the latter.
- If the router received an address list TLV from the remote PE node with a valid IP address of the CE attached to the remote PE, it not checks it against the IP address being ARPed for when replying to an ARP request over the Ethernet SAP.
- The ARP cache is flushed when the SAP bounces or when the operator manually clears the ARP cache. This results in the clearing of the CE address discovered on this SAP. However, when the SAP comes up initially or comes back up from a failure, an unsolicited ARP request is not sent over the Ethernet SAP.
- If the Ipipe service makes use of a spoke SDP, the router includes the address list TLV in the interface parameters field of the pseudowire FEC TLV in the label mapping message. The address list TLV contains the current value of the CE address in the ARP cache. If no address was learned, then an address value of 0.0.0.0 must be used.
- If the remote PE included the address list TLV in the received label mapping message, the local updates the remote PE node with the most current IP address of the Ethernet CE using a T-LDP notification message with status TLV status code is set to 0x0000002C and containing an LDP address list. The notification message is sent each time an IP address different from the current value in the ARP cache is learned. This includes when the ARP is flushed and the CE address is reset to the value of 0.0.0.0.
- If the remote PE did not include the address list TLV in the received label mapping message, the local router will not send any notification messages containing the address list TLV during the lifetime of the IP pseudowire.
- If the operator disables the **ce-address-discovery** option under the VLL service, service manager instructs LDP to withdraw the service label and the service is shutdown. The pseudowire labels will only be signaled and the service will come up if the operator re-enters the option again or enters manually the **ce-address** parameter under SAP and spoke SDP.

VLL FR SAP Procedures

The operator enables the following CE address dynamic learning procedures by enabling the **ce-address-discovery** option under the VLL service.

- Allow the service to come up without the CE address parameter configured at both the SAP and spoke SDP. If one or both parameters are configured, they are ignored.
- The operator cannot configure the **ce-address** parameter under SAP or spoke SDP when the **ce-address-discovery** option under the VLL service is enabled. Conversely, the operator is not allowed to enable the **ce-address-discovery** option under the Ipipe service if it has a SAP and/or spoke SDP with a user-entered **ce-address** parameter.
- If the router receives an invFR ARP request message over the FR SAP, it updates the ARP cache with the FR CE address. It also replies with the IP address of the CE attached to the remote PE if a valid address was advertised in the address list TLV by this remote PE. Otherwise, the router updates the ARP cache but does not reply to the invFR ARP.
- If the Ipipe service makes use of a spoke SDP, the router includes the address list TLV in the interface parameters field of the pseudowire FEC TLV in the label mapping message. The address list TLV contains the current value of the CE address in the ARP cache. If no address was learned, then an address value of 0.0.0.0 is used.
- If the remote PE included the address list TLV in the received label mapping message, the local router updates the remote PE node with the most current IP address of the FR CE using a T-LDP status notification message containing an LDP address list. The notification message is sent each time an IP address different from the current value in the ARP cache is learned. This includes when the ARP is flushed and the CE address is reset to the value of 0.0.0.00.
- If the remote PE did not include the address list TLV in the received label mapping message, the local router does not send any notification messages containing the address list TLV during the lifetime of the IP pseudowire.

VLL ATM SAP Procedures

The operator enables the following CE address dynamic learning procedures by enabling the **ce-address-discovery** option under the VLL service.

- Allow the service to come up without the **ce-address** parameter configured at both the SAP and spoke SDP. If one or both parameters are configured, they are ignored.
- The operator is not allowed to configure the **ce-address** parameter under the SAP or spoke SDP when the **ce-address-discovery** option under the VLL service is enabled. Conversely, the operator is not allowed to enable the **ce-address-discovery** option under the Ipipe service if it has a SAP and/or spoke SDP with a user-entered **ce-address** parameter.
- If the router receives an invATM ARP request message over the ATM SAP, silently discards it. The router does not support receiving or sending of an invATM ARP message.
- If the Ipipe service makes use of a spoke SDP, the router includes the address list TLV in the interface parameters field of the pseudowire FEC TLV in the label mapping message. The address list TLV contains an address value of 0.0.0.0.
- If the remote PE included the address list TLV in the received label mapping message, the local router will not make further updates to the address list TLV to the remote PE node using a T-LDP status notification message since the learned IP address of the ATM attached CE will never change away from the value of 0.0.0.0.
- If the remote PE did not include the address list TLV in the received label mapping message, the local router will not send any notification messages containing the address list TLV during the lifetime of the IP pseudowire.

VLL PPP/IPCP and Cisco-HDLC SAP Procedures

The procedures are similar to the case of an ATM SAP. The remote CE address can only be learned in the case of a PPP SAP but is not sent in the address list TLV to the remote PE in both PPP and Cisco-HDLC SAP cases.

IPv6 Support on IP Interworking VLL

The , 7750 SR, and nodes support both the transport of IPv6 packets and the interworking of IPv6 Neighbor discovery/solicitation messages on an IP Interworking VLL. IPv6 capability is enabled on an Ipipe using the **ce-address-discovery ipv6** command in the CLI.

IPv6 Datapath Operation

The IPv6 uses ICMPv6 extensions to automatically resolve IP address and link address associations. These are IP packets, as compared to ARP and invARP in IPv4, which are separate protocols and not based on IP packets. Manual configuration of IPv6 addresses is not supported on the IP Interworking VLL.

Each 7x50 PE device intercepts ICMPv6 Neighbor Discovery (RFC 2461) packets, whether received over the SAP or over the pseudowire, inspects them to learn IPv6 interface addresses and CE link-layer addresses, and modifies these packets as required according to the SAP type, and then forwards them towards the original destination. The 7x50 PE is also capable of generating packets to interwork between CEs by using IPv6 Neighbor Discovery, and CEs that use other neighbor discovery protocols to bring up the link, for example, IPv6CP for PPP.

The 7x50 PE device learns the IPv6 interface addresses for its directly-attached CE and another IPv6 interface addresses for the far-end CE. The 7x50 PE device also learns the link-layer address of the local CE and uses it when forwarding traffic between the local and far-end CEs. As with IPv4, the SAP accepts both unicast and multicast packets. For unicast packets, the 7x50 PE checks that the MAC address/IP addresses are consistent with that in the ARP cache before forwarding; otherwise the packet is silently discarded. Multicast packets are validated and forwarded. If more than one IP address is received per MAC address in a neighbor discovery packet, or if multiple neighbor discovery packets are received for a given MAC address, the currently cached address is overwritten with the most recent value.

Figure 16 illustrates the data path operation for IPv6 on an IP Interworking VLL between the Ethernet and PPP (IPv6CP) SAPs.

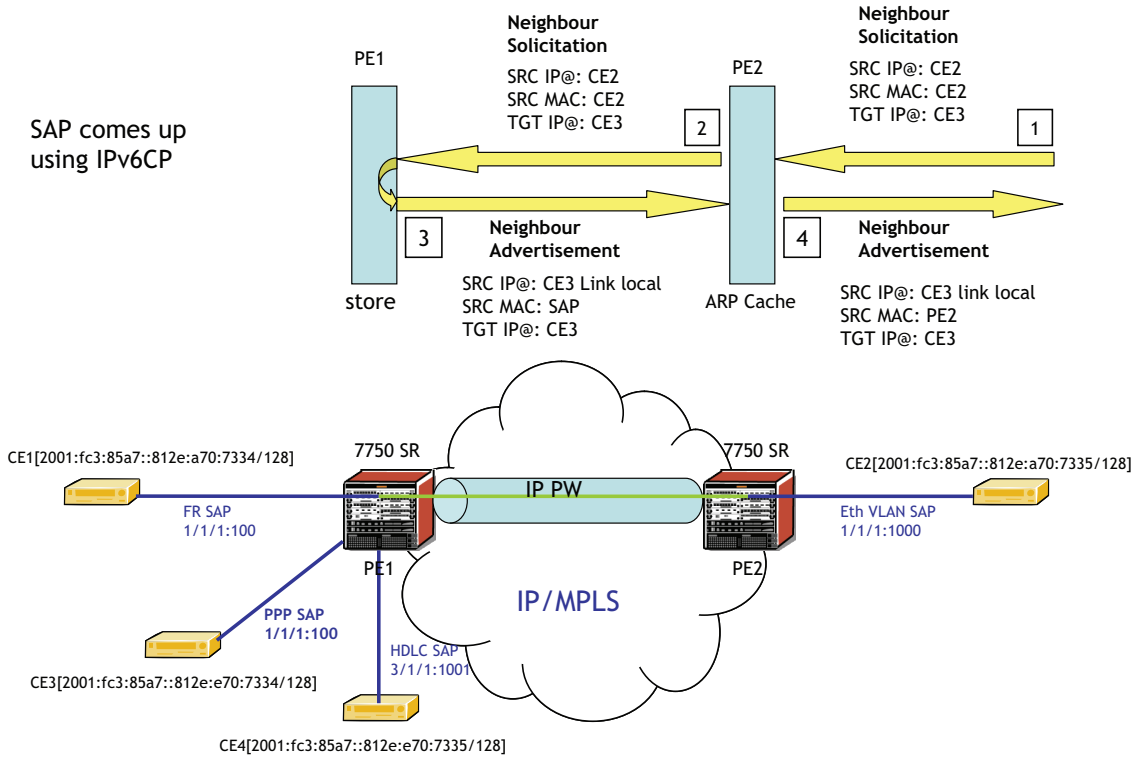


Figure 16: Data Path for Ethernet CE to PPP Attached CE

With reference to neighbor discovery between Ethernet and PPP CEs in Figure 16, the steps are as follows:

1. Ethernet attached CE2 sends a Neighbor Solicitation message towards PE2 in order to begin the neighbor discovery process.
2. PE2 snoops this message, and the MAC address and IP address of CE2 is stored in the ARP cache of PE2 before forwarding the Neighbor Solicitation on the IP pseudowire to PE1.
3. PE1 snoops this message that arrives on the IP pseudowire and stores the IP address of the remote CE2. Since CE3 is attached to a PPP SAP, which uses IPv6CP to bring up the link, PE1 generates a neighbor advertisement message and sends it on the ipipe towards PE2.
4. PE2 receives the neighbor advertisement on the Ipipe from PE1. It must replace the layer 2 address in the neighbor advertisement message with the MAC address of the SAP before forwarding to CE2.

IPv6 Stack Capability Signaling

The 7x50 supports IPv6 capability negotiation between PEs at the ends of an IP interworking VLL. Stack capability negotiation is performed if stack-capability-signaling is enabled in the CLI. Stack capability negotiation is disabled by default. In which case, it must be assumed that the remote PE supports both IPv4 and IPv6 transport over an ipipe.

A 'stack capability' sub-TLV is signaled by the two 7x50 PEs using T-LDP so that they can agree on which stacks they should be using.

By default, the IP pseudowire will always be capable of carrying IPv4 packets. Thus this capability sub-TLV is used to indicate if other stacks need to be supported concurrently with IPv4.

The stack capability sub-TLV is a part of the interface parameters of the pseudowire FEC. This means any change to the stack support requires that the pseudowire be torn down and re-signaled.

A PE that supports IPv6 on an IP pseudowire must signal the stack capability sub-TLV in the initial label mapping message for the pseudowire. For the 7x50, this means that the stack capability sub-TLV must be included if both the **stack-capability-signaling** and **ce-address-discovery ipv6** options are enabled under the VLL service.

In this release, if one PE of an IP interworking VLL supports IPv6, while the far end-PE does not support IPv6 (or ce-address-discovery ipv6 is disabled), the pseudowire does not come up.

If a 7x50 PE that supports IPv6 (that is, stack-capability-signaling ipv6 is enabled) has already sent an initial label mapping message for the pseudowire, but does not receive a ‘stack capability’ sub-TLV from the far-end PE in the initial label mapping message, or one is received but it is set to a reserved value, then the PE assumes that a configuration error has occurred. That is, if the remote PE did not include the capability sub-TLV in the received Label Mapping message, or it does include the sub-TLV but with the IPv6 bit cleared, and if stack-capability-signaling is enabled, the local 7x50 with ce-address-discovery ipv6 enabled withdraws its pseudowire label with the LDP status code “IP Address type mismatch”.

If a 7x50 PE that supports IPv6 (that is, stack-capability-signaling ipv6 is enabled) has not yet sent a label mapping message for the pseudowire and does not receive a ‘stack capability’ sub-TLV from the far-end PE in the initial label mapping message, or one is received but it is set to a reserved value, the PE assumes that a configuration error has occurred and does not send a label mapping message of its own.

If the IPv6 stack is not supported by both PEs, or at least one of the PEs does support IPv6 but does not have the **ce-address-discovery ipv6** option selected in the CLI, IPv6 packets received from the AC are discarded by the PE. IPv4 packets are always supported.

If IPv6 stack support is implemented by both PEs, but the **ce-address-discovery ipv6** command was not enabled on both so that the IP pseudowire came up with only IPv4 support, and one PE is later toggled to **ce-address-discovery ipv6**, then that PE sends a label withdraw with the LDP status code meaning “Wrong IP Address Type” (Status Code 0x0000004B9).

If the IPv6 stack is supported by both PEs, and therefore the pseudowire is established with IPv6 capability at both PEs, but the **ce-address-discovery ipv6** command on one PE is later toggled to **no ce-address-discovery ipv6** so that a PE ceases to support the IPv6 stack, then that PE sends a label withdraw with the LDP status code meaning “Wrong IP Address Type”.

Services Configuration for MPLS-TP

MPLS-TP PWs are supported in epipe, apipe and cpipe VLLs and epipe spoke termination on IES/VPRN and VPLS, iVPLS and B-VPLS.

This section describes how SDPs and spoke-sdp are used with MPLS-TP LSPs and static pseudowires with MPLS-TP OAM. It also describes how to conduct test service throughput for PWs, using lock instruct messages and loopback configuration.

MPLS-TP SDPs

Only MPLS SDPs are supported.

An SDP used for MPLS-TP supports the configuration of an MPLS-TP identifier as the far end address as an alternative to an IP address. IP addresses are used if IP/MPLS LSPs are used by the SDP, or if MPLS-TP tunnels are identified by IPv4 source / destination addresses. MPLS-TP node identifiers are used if MPLS-TP tunnels are used.

Only static SDPs with signaling off support MPLS-TP spoke-sdps.

The following CLI shows the new MPLS-TP options:.

```
config
   service
      sdp 10 [mpls | GRE | [ldp-enabled] [create]
         signaling <off | on>
         [no] lsp <xyz>
         [no] accounting-policy <policy-id>
         [no] adv-mtu-override
         [no] booking-factor <percentage>
         [no] class-forwarding
         [no] collect-stats
         [no] description <description-string>
         [no] far-end <ip-address> | [node-id {<ip-address> | <0…4,294,967,295>} [global-
id <global-id>]]
         [no] tunnel-far-end <ip-address>
         [no] keep-alive
         [no] mixed-lsp-mode
         [no] metric <metric>
         [no] network-domain <network-domain-name>
         [no] path-mtu <mtu>
         [no] pbb-etype <ethertype>
         [no] vlan-vc-etype <ethertype>
         [no] shutdown
```

The **far-end node-id** *ip-address* **global-id** *global-id* command is used to associate an SDP far end with an MPLS-TP tunnel whose far end address is an MPLS-TP node ID. If the SDP is associated with an RSVP-TE LSP, then the far-end must be a routable IPv4 address.

The system will accept the node-id being entered in either 4-octet IP address format <a.b.c.d> or unsigned integer format.

The SDP far-end refers to an MPLS-TP node-id/global-id only if:

- delivery type is MPLS
- signaling is **off**.
- keep-alive is disabled
- mixed-lsp-mode is disabled
- adv-mtu-override is disabled

An LSP will only be allowed to be configured if the far-end information matches the lsp far-end information (whether MPLS-TP or RSVP).

- Only one LSP is allowed if the far-end is an MPLS-TP node-id/global-id
- MPLS-TP or RSVP-TE LSPs are supported. However, note that LDP and BG LSPs are not blocked in CLI.

Signaling LDP or BGP is blocked if:

- far-end node-id/global-id is configured
- control-channel-status is enabled on any spoke (or mate vc-switched spoke)
- pw-path-id is configured on any spoke (or mate vc-switched spoke)
- if IES/VPRN interface spoke control-word is enabled

The following commands are blocked if a far-end node-id/global-id is configured:

- class-forwarding
- tunnel-far-end
- mixed-lsp-mode
- keep-alive
- ldp or bgp-tunnel
- adv-mtu-override

VLL Spoke SDP Configuration

The system can be a T-PE or and S-PE for a pseudowire (spoke-sdp) supporting MPLS-TP OAM. MPLS-TP related commands are applicable to spoke-sdps configured under all services supported by MPLS-TP pseudowires. All commands and functions that are applicable to spoke-sdps are supported, except for those that explicitly depend on an LDP session on the SDP or as stated below. Likewise, all existing functions on a given service SAP are supported if the spoke-sdp that it is mated to is MPLS-TP.

vc-switching is supported.

The following describes how to configure MPLS-TP on an Epipe VLL. However, a similar configuration applies to other VLL types.

A spoke-sdp bound to an SDP with the mpls-tp keyword cannot be **no shutdown** unless the ingress label, the egress label, the control word, and the pw-path-id are configured.

```
config
   service
      epipe
         [no] spoke-sdp sdp-id[:vc-id]
              [no] hash-label
              [no]standby-signaling-slave

         [no] spoke-sdp sdp-id[:vc-id] [vc-type {ether|vlan}]
             [create] [vc-switching] [no-endpoint | {endpoint [icb]}]
            egress
               vc-label <out-label>
            ingress
               vc-label <in-label>
            control-word
            bandwidth <bandwidth>
            [no] pw-path-id
               agi <agi>
               saii-type2 <global-id:node-id:ac-id>
               taii-type2 <global-id:node-id:ac-id>
               exit
            [no] control-channel-status
              [no] refresh-timer <value>
              request-timer <request-timer-secs> retry-timer <retry-timer-secs> timeout-
              multiplier <multiplier>
             no request-timer
              [no] acknowledgment
              [no] shutdown
              exit
```

The pw-path-id context is used to configure the end-to-end identifiers for an MS-PW. These may not coincide with those for the local node if the configuration is at an S-PE. The saii and taii are consistent with the source and destination of a label mapping message for a signaled PW.

The **control-channel-status** command enables static pseudowire status signaling. This is valid for any spoke-sdp where **signaling none** is configured on the SDP (for example, where T-LDP

signaling is not in use). The refresh timer is specified in seconds, from 10-65535, with a default of 0 (off). This value can only be changed if **control-channel-status** is **shutdown**. Commands that rely on PW status signaling are allowed if control-channel-status is configured for a spoke-sdp bound to an SDP with signaling off, but the system will use control channel status signaling rather than T-LDP status signaling. The ability to configure control channel status signaling on a given spoke-sdp is determined by the credit based algorithm described earlier. Control-channel-status for a particular pseudowire only counts against the credit based algorithm if it is in a **no shutdown** state and has a non-zero refresh timer and a non-zero request timer.

Note that a shutdown of a service will result in the static PW status bits for the corresponding PW being set.

The spoke-sdp is held down unless the **pw-path-id** is complete.

The system will accept the node-id of the pw-path-id saii or taii being entered in either 4-octet IP address format <a.b.c.d> or unsigned integer format.

The control-word must be enabled to use MPLS-TP on a spoke-sdp.

The optional acknowledgment to a static pw status message is enabled using the **acknowledgment** command. The default is **no acknowledgment**.

Only static pw to static pw switching is supported for MPLS-TP. Therefore, the vc-switching command is mutually exclusive with the configuration of the MPLS-TP parameters if the mate PW is not configured for an SDP with signaling off. However, vc-switching is supported if the mate SDP has signaling off.

The **pw-path-id** is only configurable if all of the following are true:

- in network mode D
- sdp signaling is off
- control-word is enabled (control-word is disabled by default)
- on service type epipe, vpls, cpipe, or IES/VPRN interface
- mate sdp signaling is off for vc-switched services
- An MPLS-TP node-id/global-id is configured under the **config>router>mpls>mpls-tp** context. This is required for OAM to provide a reply address.

In the vc-switching case, if configured on a mate spoke-sdp, then the TAII of the spoke-sdp must match the SAII of its mate, and SAII of spoke-sdp has to match the TAII of its mate.

A control-channel-status no shutdown is allowed only if all of the following are true:

- in network-mode D
- sdp signaling is off

- control-word is enabled (control-word by default is disabled)
- the service type is epipe, apipe, vpls, cpipe, or IES/VPRN interface
- mate sdp signaling is off (in vc-switched services)
- pw-status-signaling is enabled (see below)
- pw-path-id is configured for this spoke.

The **hash-label** option is only configurable if SDP far-end is not node-id/global-id.

The control channel status request mechanism is enabled when the **request-timer** <timer> parameter is non-zero. When enabled, this overrides the normal RFC-compliant refresh timer behavior. The refresh timer value in the status packet defined in RFC 6478 is always set to zero. The refresh-timer in the sending node is taken from the request-timer <timer1> timer. The two mechanisms are not compatible with each other. One node sends a request timer while the other is configured for refresh timer. In a given node, the request timer can only be configured with both acknowledgment and refresh timers disabled.

Once configured, the procedures below are used instead of the RFC 6478 procedures when a PW status changes.

The CLI commands to configure control channel status requests are shown, below:

```
[no] control-channel-status
     [no] refresh-timer <value> //0,10-65535, default:0
     [no] request-timer <timer1> retry-timer <timer2>
                  [timeout-multiplier <value>]
     [no] shutdown
     exit
```

request-timer <timer1>: 0, 10-65535, defaults: 0.

- This parameter determines the interval at which PW status messages, including a reliable delivery TLV, with the “request” bit set (see below) are sent. This cannot be enabled if refresh-timer not equal to zero (0).

retry-timer <timer2> : 3-60s

- This parameter determines the timeout interval if no response to a PW status is received. This defaults to zero (0) when no retry-timer.

timeout-multiplier <value> - 3-15.

- If a requesting node does not hear back after retry-timer times multiplier, then it must assume that the peer is down. This defaults to zero (0) when no retry-timer.

Epipe VLL Spoke-SDP Termination on IES, VPRN and VPLS

All existing commands (except for those explicitly specified below) are supported for spoke-sdp termination on IES, VPRN and VPLS (VPLS, iVPLS and bVPLS and routed VPLS) services. In addition, the MPLS-TP commands listed above are supported. The syntax and default values, and functional behavior of these commands is the same as for Epipe VLLs, as specified above.

In addition, the PW Control Word is supported on spoke-sdp termination on IES/VPRN interfaces for pseudowires of type “Ether” with statically assigned labels (signaling off) for spoke-sdps configured with MPLS-TP Identifiers.

The following CLI commands under spoke-sdp are blocked for spoke-sdps with statically assigned labels (and the SDP has signaling off) and MPLS-TP identifiers:

- **no status-signaling** – This command causes the spoke-sdp to fall back to using PW label withdrawal as a status signaling method. However, T-LDP is not supported on MPLS-TP SDPs. Control channel status signaling should always be used for signaling PW status. Note that since active/standby dual-homing into a routed VPLS requires the use of T-LDP label withdrawal as the method for status signaling, active/standby dual-homing into routed VPLS is not supported if the spoke-sdps are MPLS-TP.
- **propagate-mac-flush** – This command requires the ability to receive MAC Flush messages using T-LDP signaling and is blocked.

Configuring MPLS-TP Lock Instruct and Loopback

MPLS-TP supports lock instruct and loopback for PWs. The topics in this section are:

MPLS-TP PW Lock Instruct and Loopback Overview

The lock instruct and loopback capability for MPLS-TP PWs includes the ability to:

- administratively lock a spoke-sdp with MPLS-TP identifiers
- divert traffic to and from an external device connected to a SAP
- create a data path loopback on the corresponding PW at a downstream S-PE or T-PE that was not originally bound to the spoke-sdp being tested
- forward test traffic from an external test generator into an administratively locked PW, while simultaneously blocking the forwarding of user service traffic

MPLS-TP provides the ability to conduct test service throughput for PWs, using lock instruct messages and loopback configuration. To conduct a service throughput test, you can apply an administrative lock at each end of the PW. This creates a test service, that contains the SAP connected to the external device. Lock request messaging is not supported. You can also configure a MEP to send a lock instruct message to the far-end MEP. The lock instruct message is carried in a G-ACh on Channel 0x0026. A lock can be applied using the CLI or NMS. The forwarding state of the PW can be either active or standby.

After locking a PW, you can put it into loopback mode (for two way tests) so the ingress data path in the forward direction is cross connected to the egress data path in the reverse direction of the PW. This is accomplished by configuring the source MEP to send a loopback request to an intermediate MIP or MEP. A PW loopback is created at the PW level, so everything under the PW label is looped back. This distinguishes a PW loopback from a service loopback, where only the native service packets are looped back. The loopback is also configured through CLI or NMS.

The following MPLS-TP lock instruct and loopback functionality is supported:

- An MPLS-TP loopback can be created for an epipe, cpipe or apipe VLL

- Test traffic can be inserted at an epipe, cpipe or apipe VLL endpoint or at an Epipe spoke-sdp termination on a VPLS interface

Lock PW End-Point Model

You can administratively lock a spoke-sdp by locking the host service using the **admin-lock** parameter of the **tools** command. The following conditions and constraints apply:

- Both ends of a PW or MS-PW represented by a spoke-sdp must be administratively locked.
- Test traffic can be injected into the spoke-sdp using a SAP defined within a test service. The test service must be identified in the `tools` command at one end of the locked PW.
- All traffic is forwarded to and from the test SAP defined in the test service, which must be of a type that is compatible with the spoke-sdp.
- Traffic to and from a non-test-SAP is dropped. If no test SAP is defined all traffic received on the spoke-SDP is dropped, and all traffic received on the paired SAP is also dropped.
- If a spoke-sdp is administratively locked, it is treated as operationally down. If a VLL SAP is paired with a spoke-sdp that is administratively locked, the SAP OAM treats this as if the spoke-sdp is operationally down.
- If a VPLS interface is paired to a spoke-sdp that is administratively locked, the L2 interface is taken down locally.
- Control-channel-status must be shutdown prior to administratively locking a spoke-sdp.

PW Redundancy and Lock Instruct and Loopback

It is possible to apply an administrative lock and loopback to one or more spoke-sdps within a redundant set. That is, it is possible to move a spoke-sdp from an existing endpoint to a test service. When an administrative lock is applied to a spoke-sdp, it becomes operationally down and cannot send or receive traffic from the normal service SAP or spoke interface. If the lock is applied to all the spoke-sdps in a service, then all the spoke-sdps will become operationally down.

Configuring a Test SAP for an MPLS-TP PW

A test SAP is configured under a unique test service type. This looks similar to a normal service context, but will normally only contain a SAP configuration.

```
config
     service
          epipe <service-id> [test][create]
               [no] sap <sap-id>
               [no] shutdown
          [no] shutdown
config
     service
          apipe <service-id> [vc-type {atm-vcc | atm-sdu | atm-vpc | atm-cell}
           [test][create]
               [no] sap <sap-id>
               [no] shutdown
          [no] shutdown
config
     service
          cpipe <service-id> [vc-type {satop-e1 | satop-t1 | cesopsn |  cesopsn-cas}
[test][create]
               [no] sap <sap-id>
               [no] shutdown
          [no] shutdown
```

You can define test SAPs appropriate to any service or PW type supported by MPLS-TP including an apipe, cpipe or epipe. The following test SAP types are supported:

- Ethernet NULL, .1q, Q-in-Q
- ATM VC, VP, VT and so on
- TDM E1, E3, DS0, DS3 and so on

The following constraints and conditions apply:

- Up to a maximum a 16 test services can be configured per system.
- It is possible to configure access ingress and access egress QoS policies on a test SAP, as well as any other applicable SAP-specific commands and overrides.
- Vc-switching and spoke-sdp are blocked for services configured under the test context.
- The **test** keyword is mutually exclusive with vc-switching and customer.
- Valid commands under a compatible test service context do not need to be blocked just because the service is a test service.

Configuring an Administrative Lock

An administrative lock is configured on a spoke-sdp using the **admin-lock** option of the **tools perform** command, as follows:

```
tools
     perform
          service-id <svc-id>
               admin-lock
                    pw
                         sdp <sdp-id> admin-lock [test-svc-id <id>]
```

The following conditions and constraints apply for configuring an administrative lock:

- Can be configured either on a spoke-sdp that is bound to a SAP, another spoke-sdp or a VPLS interface.
- Is only allowed if a PW path ID is defined (for example, for static PWs with MPLS-TP identifiers).
- Cannot be configured on spoke-sdps that are an ICB or if the vc-switching keyword is present.
- The control-channel-status must be shutdown. The operator should also shutdown control-channel-status on spoke-sdps belonging to an MS-PW at an S-PE whose far ends are administratively locked at its T-PEs. This should be enforced throughout the network management if using the Service Access Manager.
- When enabled, all traffic on the spoke-sdp is sent to and from a paired SAP that has the **test** keyword present, if such a SAP exists in the X endpoint. Otherwise all traffic to and from the paired SAP is dropped.
- Can be configured at a spoke-sdp that is bound to a VLL SAP or a VPLS interface.
- The **test-svc-id** parameter refers to the test service that should be used to inject test traffic into the service. The test service must be of a compatible type to the existing spoke-sdp under test (see Table 8).
- If the **test-svc-id** parameter is not configured on an admin-locked spoke-sdp, then user traffic is simply blocked on the spoke-sdp.

The service manager should treat an administrative lock as a fault from the perspective of a paired SAP that is not a test SAP. This will cause the appropriate SAP OAM fault indication.

Table 8 illustrates the mapping between supported real services and their corresponding test services.

Table 8: Mapping of Real Services to Test Service Types

Service	Test Service
CPIPE	CPIPE
EPIPE	EPIPE
APIPE	APIPE
VPLS	EPIPE
PBB VPLS	EPIPE

Configuring a Loopback

If a loopback is configured on a spoke-sdp, then all traffic on the ingress direction of the spoke-sdp and associated with the ingress vc-label is forwarded to the egress direction of the spoke-sdp. A loopback may be configured at either a T-PE or an S-PE. Note that it is recommended that you configure an administrative lock before configuring the loopback on a spoke-sdp. This is enforced by the NMS.

A data path loopback is configured using a tools perform command, as follows:

```
tools
     perform
          service-id <svc-id>
               loopback
                    pw
                         sdp <sdp-id>:<vc-id> {start | stop}
```

The following constraints and conditions apply for PW loopback configuration:

- The spoke-sdp cannot be an ICB or be bound to a VPLS interface.
- A PW path ID must be configured, that is, the spoke-sdp must be static and use MPLS-TP identifiers.
- The spoke-sdp must be bound to a VLL mate SAP or another spoke-sdp that is not an ICB.
- The control-channel-status must be shutdown.
- The following is disabled on a spoke-sdp for which a loopback is configured:
  - → Filters
  - → PW shaping
- Only network port QoS is supported.

VCCV BFD support for VLL, Spoke-SDP Termination on IES and VPRN, and VPLS Services

Topics include:

VCCV BVD Support

The SR OS supports RFC 5885, which specifies a method for carrying BFD in a pseudowire-associated channel. This enables BFD to monitor the pseudowire between its terminating PEs, irrespective of how many P routers or switching PEs the pseudowire may traverse. This makes it possible for faults that are local to individual pseudowires to be detected, whether or not they also affect forwarding for other pseudowires, LSPs or IP packets. VCCV BFD is ideal for monitoring specific high-value services, where detecting forwarding failures (and potentially restoring from them) in the minimal amount of time is critical.

VCCV BFD is supported on VLL services using T-LDP spoke-SPDs or BGP VPWS. It is supported for Apipe, Cpipe, Epipe, Fpipe, and Ipipe VLL services.

VCCV BFD is supported on IES/VPRN services with T-LDP spoke -SDP termination (for Epipes and Ipipes).

VCCV BFD is supported on LDP- and BGP-signaled pseudowires, and on pseudowires with statically configured labels, whether signalling is off or on for the SDP. VCCV BFD is not supported on MPLS-TP pseudowires

VCCV BFD is supported on VPLS services (both spoke-SDPs and mesh-SDPs). VCCV BFD is configured by:

- configuring generic BFD session parameters in a BFD template.
- applying the BFD template to a spoke-SDP or pseudowire-template binding, using the **bfd-template** *template_name* command.
- enabling the template on that spoke-SDP, mesh-SDP or pseudowire-template binding using the **bfd-enable** command.

VCCV BFD Encapsulation on a Pseudowire

The SR OS supports IP/UDP encapsulation for BFD. With this encapsulation type, the UDP headers are included on the BFD packet. IP/UDP encapsulation is supported for pseudowires that use router alert (VCCV Type 2), and for pseudowires with a control word (VCCV Type 1). In the control word case, the IPv4 channel (channel type 0x0021) is used. On the 7x50, the destination IPv4 address is fixed at 127.0.0.1 and the source address is 127.0.0.2.

VCCV BFD sessions run end-to-end on a switched pseudowire. They do not terminate on an intermediate S-PE; therefore, the TTL of the pseudowire label on VCCV BFD packets is always set to 255 to ensure that the packets reach the far-end T-PE of an MS-PW.

BFD Session Operation

BFD packets flow along the full length of a PW, from T-PE to T-PE. Since they are not intercepted at an S-PE, single-hop initialization procedures are used.

A single BFD session exists per pseudowire.

BFD runs in asynchronous mode.

BFD operates as a simple connectivity check on a pseudowire. The BFD session state is reflected in the MIBs and in the **show>service id>sdp>vccv-bfd session** command. In this sense, BFD operates in a similar manner to other proactive OAM tools, such as SAA with VCCV Ping. BFD is not used to change the operation state of the pseudowire or to modify pseudowire redundancy. Furthermore, mapping the BFD state to SAP OAM is not supported.

VCCV BFD runs in software with a minimum supported timer interval of 1s.

Note that BFD is only used for fault detection. While RFC 5885 provides a mode in which VCCV BFD can be used to signal pseudowire status, this mode is only applicable for pseudowires that have no other status signaling mechanism in use. LDP status and static pseudowire status signaling always take precedence over BFD-signaled PW status, and BFD-signaled pseudowire status is not used on pseudowires that use LDP status or static pseudowire status signaling mechanisms.

Configuring VCCV BFD

Generic BFD session parameters are configured for VCCV using the **bfd-template** command, in the **config>router>bfd** context. However, there are some restrictions.

For VCCV, the BFD session can not terminate on the CPM network processor. Therefore, an error is generated if the user tries to bind a BFD template using the **type cpm-np** command within the **config>router>bfd>bfd-template** context.

As well, the minimum supported value for the **transmit-interval** and **receive-interval** commands when BFD is used for VCCV-BFD is 1s. Attempting to bind a BFD template with any unsupported transmit or receive interval will generate an error.

Finally, attempting to commit changes to a BFD template that is already bound to a pseudowire where the new values are invalid for VCCV BFD will result in an error.

Note that if the above BFD timer values are changed in a given template, any BFD sessions on pseudowires to which that template is bound will try to renegotiate their timers to the new values.

Commands within the BFD-template use a **begin-commit** model. To edit any value within the BFD template, a **begin** command needs to be executed once the template context has been entered. However, a value will still be stored temporarily in the template-module until the **commit** command is issued. Once the **commit** is issued, values will be used by other modules such as the MPLS-TP module and BFD module.

For pseudowires where the pseudowire template does not apply (for example, LDP-signaled spoke-SDPs for a VLL service that uses the pseudowire ID FEC (FEC128) or spoke-SDPs with static pseudowire labels with or without MPLS-TP identifiers), a named BFD template is configured on the spoke-SDP using the command **config>service>epipe | cpipe | apipe | fpipe | ipipe>spoke-sdp>bfd-template** *name* and then enabled using the command **config>service> epipe | cpipe | apipe | fpipe | ipipe>spoke-sdp>bfd-enable**.

Configuring and enabling a BFD template on a static pseudowire already configured with MPLS-TP identifiers (that is, with a pw-path-id) or on a spoke-SDP with a configured pw-path-id is not supported. Likewise, if a BFD template is configured and enabled on a spoke-SDP, then a pw-path-id can not be configured on the spoke-SDP.

The **bfd-enable** command is blocked on a spoke-SDP configured with VC-switching. This is because VCCV BFD always operates end-to-end on an MS-pseudowire. It is not possible to extract VCCV BFD packets at the S-PE

For IES and VPRN spoke-SDP termination where the pseudowire template does not apply (that is, where the spoke-SDP is signaled with LDP and uses the pseudowire ID FEC (FEC128), the BFD template is configured using the command **config>service>ies | vprn>interface>spoke-sdp>bfd-template** *name* and then enabled using the command **config>service>ies | vprn>interface>spoke-sdp>bfd-enable**.

For H-VPLS where the PW-Template does not apply (i.e LDP-VPLS spoke and mesh-sdps that use the Pwid FEC(FEC128) the bfd template is configured using config>service>vpls>spoke>sdp>bfd-name *name* or config>service>vpls>mesh-sdp>bfd-name *name*. VCCV BFD is then enabled with the bfd-enable command under the VPLS spoke-sdp or mesh-sdp context.

Pseudowires where the pw-template does apply and that support VCCV BFD are as follows:

- BGP-AD, which is signaled using the Generalised pseudowire ID FEC (FEC129) with AII type I
- BGP VPLS
- BGP VPWS

For these pseudowire types, a named BFD template is configured and enabled from the pseudowire template binding context.

For BGP VPWS, the BFD template is configured using the command **config>service>epipe>bgp>pw-template-binding>bfd-template** *name* and then enabled using the command **config>service>epipe>bgp>pw-template-binding>bfd-enable**.

Pseudowire Switching

The pseudowire switching feature provides the user with the ability to create a VLL service by cross-connecting two spoke SDPs. This feature allows the scaling of VLL and VPLS services in a large network in which the otherwise full mesh of PE devices would require thousands of Targeted LDP (T-LDP) sessions per PE node.

Services with one SAP and one spoke SDP are created normally on the PE; however, the target destination of the SDP is the pseudowire switching node instead of what is normally the remote PE. In addition, the user configures a VLL service on the pseudowire switching node using the two SDPs.

The pseudowire switching node acts in a passive role with respect to signalling of the pseudowires. It waits until one or both of the PEs sends the label mapping message before relaying it to the other PE. This is because it needs to pass the Interface Parameters of each PE to the other.

A pseudowire switching point TLV is inserted by the switching pseudowire to record its system address when relaying the label mapping message. This TLV is useful in a few situations:

- It allows for troubleshooting of the path of the pseudowire especially if multiple pseudowire switching points exist between the two PEs.
- It helps in loop detection of the T-LDP signalling messages where a switching point would receive back a label mapping message it had already relayed.
- The switching point TLV is inserted in pseudowire status notification messages when they are sent end-to-end or from a pseudowire switching node towards a destination PE.

Pseudowire OAM is supported for the manual switching pseudowires and allows the pseudowire switching node to relay end-to-end pseudowire status notification messages between the two PEs. The pseudowire switching node can generate a pseudowire status and to send it to one or both of the PEs by including its system address in the pseudowire switching point TLV. This allows a PE to identify the origin of the pseudowire status notification message.

In the Figure 17, the user configures a regular Epipe VLL service PE1 and PE2. These services consist each of a SAP and a spoke SPD. However, the target destination of the SDP is actually not the remote PE but the pseudowire switching node. In addition, the user configures an Epipe VLL service on the pseudowire switching node using the two SDPs.

Figure 17: Pseudowire Service Switching Node

Configuration examples can be found in Configuring Two VLL Paths Terminating on T-PE2 on page 169.

Pseudowire Switching with Protection

Pseudowire switching scales VLL and VPLS services over a multi-area network by removing the need for a full mesh of targeted LDP sessions between PE nodes. Figure 18 illustrates the use of pseudowire redundancy to provide a scalable and resilient VLL service across multiple IGP areas in a provider network.

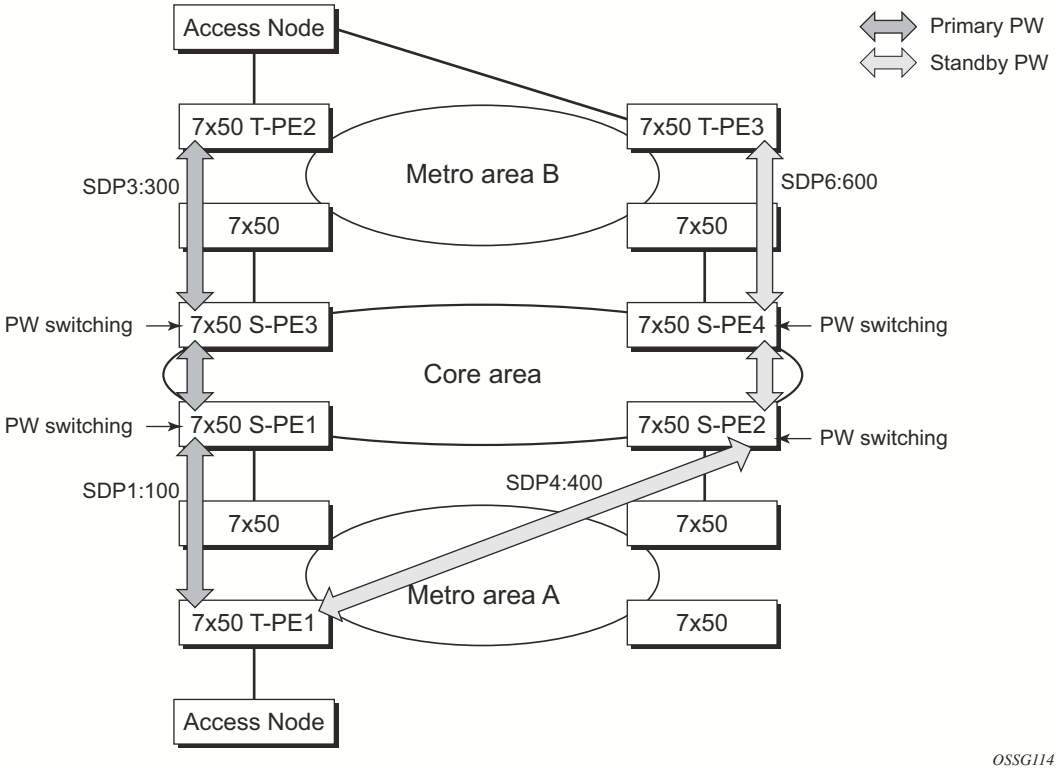


Figure 18: VLL Resilience with Pseudowire Redundancy and Switching

In the network in Figure 18, PE nodes act as masters and pseudowire switching nodes act as slaves for the purpose of pseudowire signaling. A switching node will need to pass the SAP Interface Parameters of each PE to the other.T-PE1 sends a label mapping message for the Layer 2 FEC to the peer pseudowire switching node" for example, S-PE1. It will include the SAP interface parameters, such as MTU, in the label mapping message. S-PE1 checks the FEC against the local information and if a match exists, it appends the optional pseudowire switching point TLV to the FEC TLV in which it records its system address. T-PE1 then relays the label mapping message to S-PE2. S-PE2 performs similar operations and forwards a label mapping message to T-PE2. The same procedures are followed for the label mapping message in the reverse direction, for example, from T-PE2 to T-PE1. S-PE1 and S-PE2 will effect the spoke SDP cross-connect only when both directions of the pseudowire have been signaled and matched.

The pseudowire switching TLV is useful in a few situations. First, it allows for troubleshooting of the path of the pseudowire especially if multiple pseudowire switching points exist between the two T-PE nodes. Secondly, it helps in loop detection of the T-LDP signaling messages where a switching point receives back a label mapping message it already relayed. Finally, it can be inserted in pseudowire status messages when they are sent from a pseudowire switching node towards a destination PE.

Pseudowire status messages can be generated by the T-PE nodes and/or the S-PE nodes. Pseudowire status messages received by a switching node are processed and then passed on to the next hop. An S-PE node appends the optional pseudowire switching TLV, with its system address added to it, to the FEC in the pseudowire status notification message only if it originated the message or the message was received with the TLV in it. Otherwise, it means the message was originated by a T-PE node and the S-PE should process and pass the message without changes except for the VCID value in the FEC TLV.

Pseudowire Switching Behavior

In the network in Figure 18, PE nodes act as masters and pseudowire switching nodes act as slaves for the purpose of pseudowire signaling. This is because a switching node will need to pass the SAP interface parameters of each PE to the other.T-PE1 sends a label mapping message for the Layer 2 FEC to the peer pseudowire switching node, for example, S-PE1. It will include the SAP interface parameters, such as MTU, in the label mapping message. S-PE1 checks the FEC against the local information and if a match exists, it appends the optional pseudowire switching point TLV to the FEC TLV in which it records its system address. T-PE1 then relays the label mapping message to S-PE2. S-PE2 performs similar operation and forwards a label mapping message to T-PE2. The same procedures are followed for the label mapping message in the reverse direction, for example, from T-PE2 to T-PE1. S-PE1 and S-PE2 will effect the spoke SDP cross-connect only when both directions of the pseudowire have been signaled and matched.

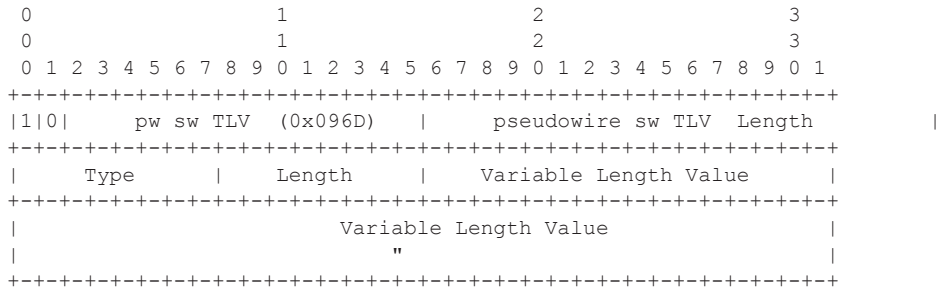
Pseudowire status notification messages can be generated by the T-PE nodes and/or the S-PE nodes. Pseudowire status notification messages received by a switching node are processed and then passed on to the next hop. An S-PE node appends the optional pseudowire switching TLV, with its system address added to it, to the FEC in the pseudowire status notification message only if it originated the message or the message was received with the TLV in it. Otherwise, it means the message was originated by a T-PE node and the S-PE should process and pass the message without changes except for the VC ID value in the FEC TLV.

The merging of the received T-LDP status notification message and the local status for the spoke SDPs from the service manager at a PE complies with the following rules:

- When the local status for both spokes is up, the S-PE passes any received SAP or SDP-binding generated status notification message unchanged, for example, the status notification TLV is unchanged but the VC-ID in the FEC TLV is set to value of the pseudowire segment to the next hop.
- When the local operational status for any of the spokes is down, the S-PE always sends SDP-binding down status bits regardless if the received status bits from the remote node indicated SAP up/down or SDP-binding up/down.

Pseudowire Switching TLV

The format of the pseudowire switching TLV is as follows:



PW sw TLV Length — Specifies the total length of all the following pseudowire switching point TLV fields in octets

Type — Encodes how the Value field is to be interpreted.

Length — Specifies the length of the Value field in octets.

Value — Octet string of Length octets that encodes information to be interpreted as specified by the Type field.

Pseudowire Switching Point Sub-TLVs

Below are details specific to pseudowire switching point sub-TLVs:

pseudowire ID of last pseudowire segment traversed — This sub-TLV type contains a pseudowire ID in the format of the pseudowire ID

Pseudowire switching point description string — An optional description string of text up to 80 characters long.

IP address of pseudowire switching point.

The IP V4 or V6 address of the pseudowire switching point. This is an optional sub-TLV.

MH VCCV capability indication.

Static-to-Dynamic Pseudowire Switching

When one segment of the pseudowire cross-connect at the S-PE is static while the other is signaled using T-LDP, the S-PE operates much like a T-PE from a signaling perspective and as an S-PE from a data plane perspective.

The S-PE signals a label mapping message as soon as the local configuration is complete. The control word C-bit field in the pseudowire FEC is set to the value configured on the static spoke-sdp.

When the label mapping for the egress direction is also received from the T-LDP peer, and the information in the FEC matches that of the local configuration, the static-to-dynamic cross-connect is effected.

Note that it is possible that end nodes of a static pseudowire segment be misconfigured. In this case, an S-PE or T-PE node may be receiving packets with the wrong encapsulation. In this case, it is possible that an invalid payload will be forwarded over the pseudowire or the SAP respectively. Furthermore, if the S-PE or T-PE node is expecting the control word in the packet encapsulation and the received packet comes with no control word but the first nibble below the label stack is 0x0001, the packet may be mistaken for a VCCV OAM packet and may be forwarded to the CPM. In that case, the CPM will perform a check of the IP header fields such as version, IP header length, and checksum. If any of this fails the VCCV packet will be discarded.

Ingress VLAN Swapping

This feature is supported on VPLS and VLL services where the end to end solution is built using two node solutions (requiring SDP connections between the nodes).

In VLAN swapping, only the VLAN-id value is copied to the inner VLAN position. The Ethertype of the inner tag will be preserved and all consecutive nodes will work with that value. Similarly, the dot1p bits value of outer-tag will not be preserved.

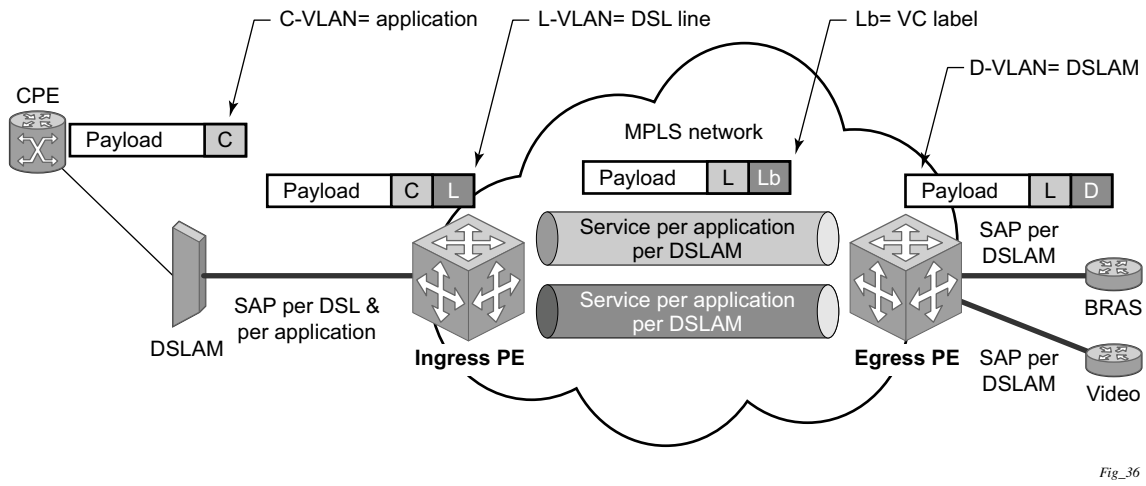


Figure 19: Ingress VLAN Swapping

The network diagram in Figure 19 describes the network where at user access side (DSLAM facing SAPs) every subscriber is represented by several QinQ SAPs with inner-tag encoding service and outer-tag encoding subscriber (DSL line). The aggregation side (BRAS or PE facing SAPs) the is represented by DSL line number (inner VLAN tag) and DSLAM (outer VLAN tag). The effective operation on VLAN tag is to “drop inner tag at access side and push another tag at the aggregation side”.

Ingress VLAN Translation

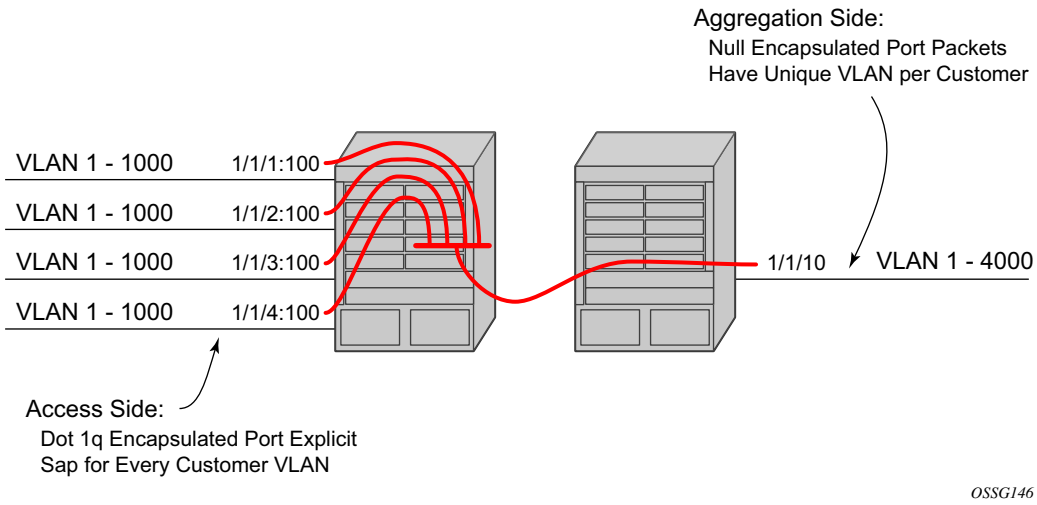


Figure 20: Ingress VLAN Translation

The drawing in Figure 20 indicates an application where different circuits are aggregated in the VPLS-based network.The access side is represented by an explicit do1q encapsulated SAP. As the VLAN-id is port specific, those connected to different ports might have the same VLAN. The aggregation side (the right side Figure 20) is aggregated on the same port, and hence, unique a VLAN-id is required.

Pseudowire Redundancy

Pseudowire redundancy provides the ability to protect a pseudowire with a pre-provisioned pseudowire and to switch traffic over to the secondary standby pseudowire in case of a SAP and/or network failure condition. Normally, pseudowires are redundant by the virtue of the SDP redundancy mechanism. For instance, if the SDP is an RSVP LSP and is protected by a secondary standby path and/or by Fast-Reroute paths, the pseudowire is also protected. However, there are a couple of applications in which SDP redundancy does not protect the end-to-end pseudowire path:

- There are two different destination PE nodes for the same VLL service. The main use case is the provision of dual-homing of a CPE or access node to two PE nodes located in different POPs. The other use case is the provision of a pair of active and standby BRAS nodes, or active and standby links to the same BRAS node, to provide service resiliency to broadband service subscribers.
- The pseudowire path is switched in the middle of the network and the SR-Series pseudowire switching node fails.

Pseudowire and VPLS link redundancy extends link-level resiliency for pseudowires and VPLS to protect critical network paths against physical link or node failures. These innovations enable the virtualization of redundant paths across the metro or core IP network to provide seamless and transparent fail-over for point-to-point and multi-point connections and services. When deployed with multi-chassis LAG, the path for return traffic is maintained through the pseudowire or VPLS switchover, which enables carriers to deliver “always on” services across their IP/MPLS networks.

Dynamic Multi-Segment Pseudowire Routing

Overview

Dynamic Multi-Segment Pseudowire Routing (Dynamic MS-PWs) enable a complete multi-segment pseudowire to be established, while only requiring per-pseudowire configuration on the T-PEs. No per-pseudowire configuration is required on the S-PEs. End-to-end signaling of the MS-PW is achieved using T-LDP, while multi-protocol BGP is used to advertise the T-PEs, so allowing dynamic routing of the MS-PW through the intervening network of S-PEs. Dynamic multi-segment pseudowires are described in the IETF in draft-ietf-pwe3-dynamic-ms-pw-13.txt.

Figure 21 illustrates the operation of dynamic MS-PWs.

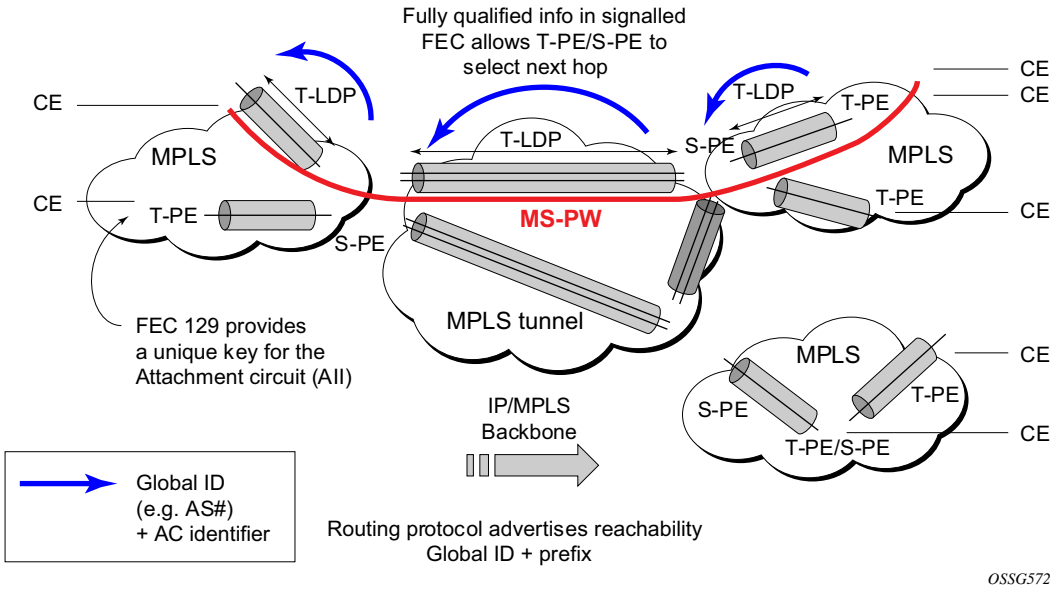


Figure 21: Dynamic MS-PW Overview

The FEC 129 AII Type 2 structure depicted in Figure 22 is used to identify each individual pseudowire endpoint:

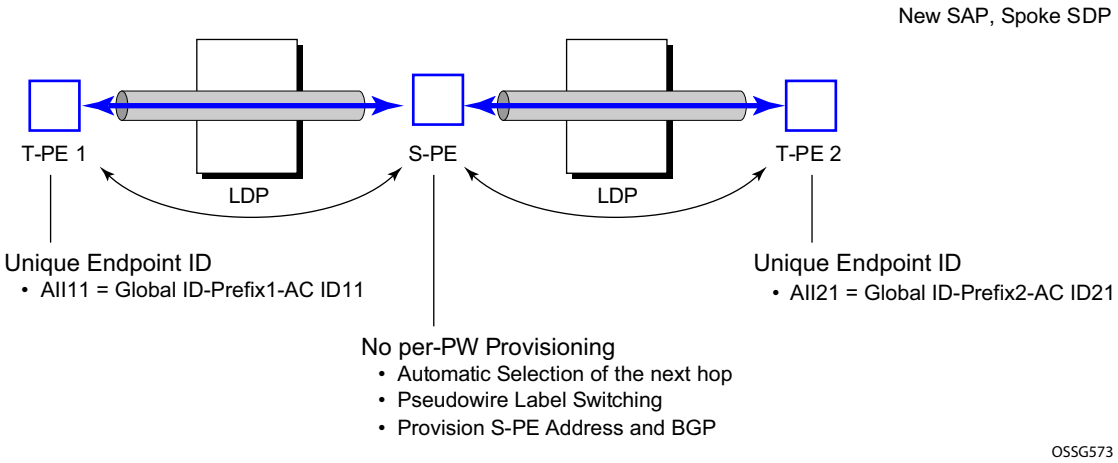


Figure 22: MS-PW Addressing using FEC129 AII Type 2

A 4-byte global ID followed by a 4 byte prefix and a 4 byte attachment circuit ID are used to provide for hierarchical, independent allocation of addresses on a per service provider network basis. The first 8 bytes (Global ID + Prefix) may be used to identify each individual T-PE or S-PE as a loopback Layer 2 Address.

This new AII type is mapped into the MS-PW BGP NLRI (a new BGP AFI of L2VPN, and SAFI for network layer reachability information for dynamic MS-PWs. As soon as a new T- PE is configured with a local prefix address of global id:prefix, pseudowire routing will proceed to advertise this new address to all the other T- PEs and S-PEs in the network, as depicted in Figure 23:

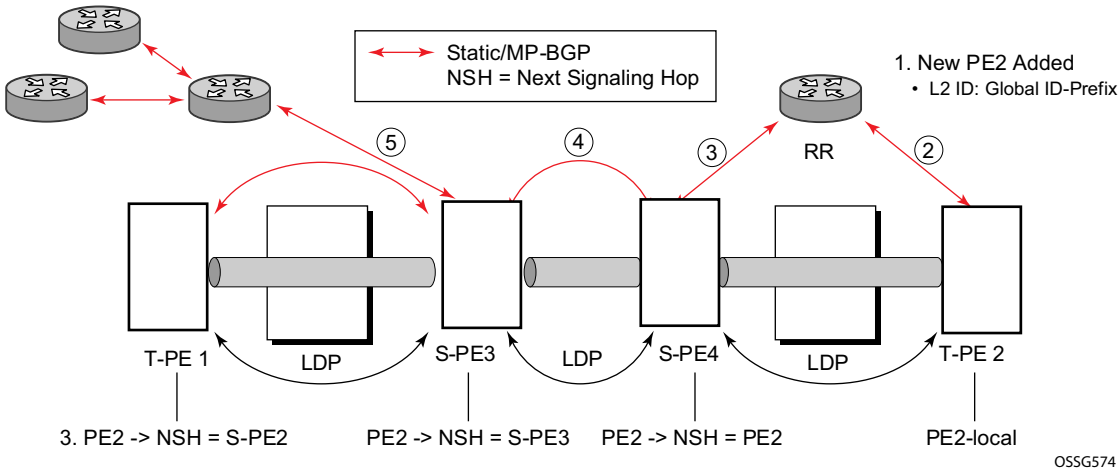


Figure 23: Advertisement of PE Addresses by PW Routing

In step 1 a new T-PE (T-PE2) is configured with a local prefix.

Next, in steps 2-5, MP-BGP will use the NLRI for the MS-PW routing SAFI to advertise the location of the new T-PE to all the other PEs in the network. Alternatively, static routes may be configured on a per T-PE/S-PE basis to accommodate non-BGP PEs in the solution.

As a result, pseudowire routing tables for all the S-PEs and remote T-PEs are populated with the next hop to be used to reach T-PE2.

VLL services can then be established, as illustrated in Figure 24.

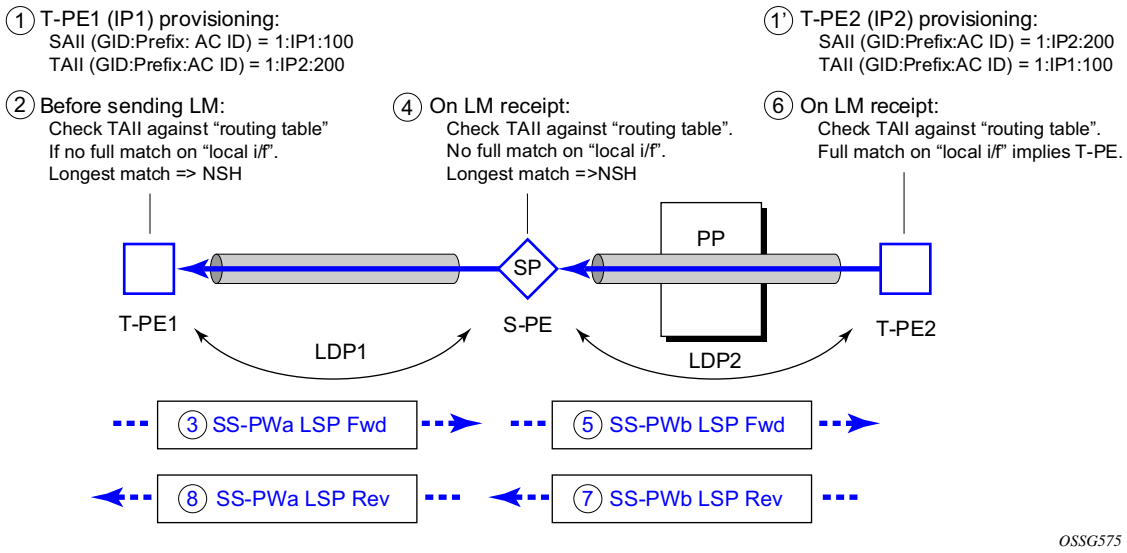


Figure 24: Signaling of Dynamic MS-PWs using T-LDP

In step 1 and 1' the T-PEs are configured with the local and remote endpoint information, Source AII (SAII), Target AII (TAII). On the 7x50, the AIIs are locally configured for each spoke SDP, according to the model shown in Figure 25. The 7x50 therefore provides for a flexible mapping of AII to SAP. That is, the values used for the AII are through local configuration, and it is the context of the spoke SDP that binds it to a specific SAP.

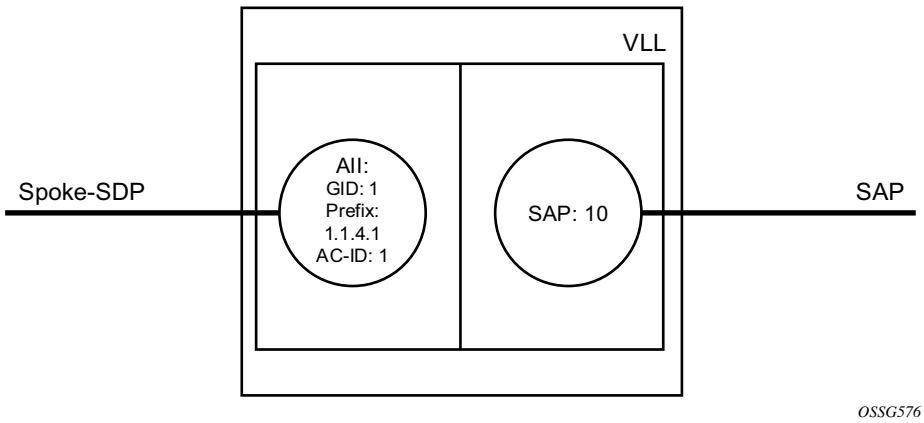


Figure 25: Mapping of AII to SAP

Before T-LDP signaling starts, the two T-PEs decide on an active and passive relationship using the highest AII (comparing the configured SAII and TAII) or the configured precedence. Next, the active T-PE (in the IETF draft this is referred to as the source T-PE or ST-PE) checks the PW Routing Table to determine the next signaling hop for the configured TAII using the longest match between the TAII and the entries in the PW routing table

This signaling hop is then used to choose the T-LDP session to the chosen next-hop S-PE. Signaling proceeds through each subsequent S-PE using similar matching procedures to determine the next signaling hop. Otherwise, if a subsequent S-PE does not support dynamic MS-PW routing and thus uses a statically configured PW segment, the signaling of individual segments follows the procedures already implemented in the PW Switching feature. Note that BGP can install a PW AII route in the PW routing table with ECMP next-hops. However when LDP needs to signal a PW with matching TAII, it will choose only one next-hop from the available ECMP next-hops. PW routing supports up to 4 ECMP paths for each destination.

The signaling of the forward path ends once the PE matches the TAII in the label mapping message with the SAII of a spoke SDP bound to a local SAP. The signaling in the reverse direction can now be initiated, which follows the entries installed in the forward path. The PW Routing tables are not consulted for the reverse path. This ensures that the reverse direction of the PW follows exactly the same set of S-PEs as the forward direction.

This solution can be used in either a MAN-WAN environment or in an Inter-AS/Inter-Provider environment as depicted in Figure 26.

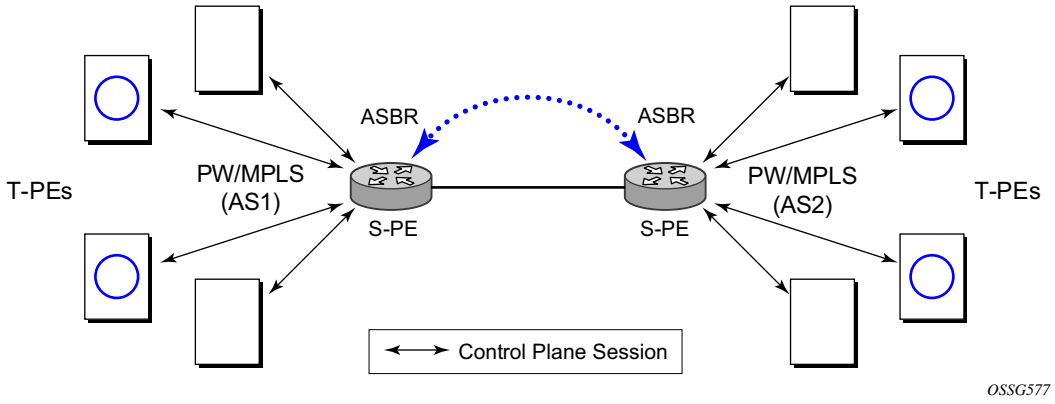


Figure 26: VLL Using Dynamic MS-PWs, Inter-AS Scenario

Note that data plane forwarding at the S-PEs uses pseudowire service label switching, as per the pseudowire switching feature.

Pseudowire Routing

Each S-PE and T-PE has a pseudowire routing table that contains a reference to the T-LDP session to use to signal to a set of next hop S-PEs to reach a given T-PE (or the T-PE if that is the next hop). For VLLs, this table contains aggregated AII Type 2 FECs and may be populated with routes that are learned through MP-BGP or that are statically configured.

MP-BGP is used to automatically distribute T-PE prefixes using the new MS-PW NLRI, or static routes can be used. The MS-PW NLRI is composed of a Length, an 8-byte RD, a 4-byte Global-ID, a 4-byte local prefix, and (optionally) a 4-byte AC-ID. Support for the MS-PW address family is configured in CLI under **config>router>bgp>family ms-pw**.

MS-PW routing parameters are configured in the **config>service>pw-routing** context.

In order to enable support for dynamic MS-PWs on a 7x50 node to be used as a T-PE or S-PE, a single, globally unique, S-PE ID, known as the S-PE Address, is first configured under **config>service>pw-routing** on each 7x50 to be used as a T-PE or S-PE. The S-PE Address has the format global-id:prefix. It is not possible to configure any local prefixes used for pseudowire routing or to configure spoke SPDs using dynamic MS-PWs at a T-PE unless an S-PE address has already been configured. The S-PE address is used as the address of a node used to populate the switching point TLV in the LDP label mapping message and the pseudowire status notification sent for faults at an S-PE.

Each T-PE is also be configured with the following parameters:

a. Global ID — This is a 4 byte identifier that uniquely identifies an operator or the local network.

b. Local Prefix — One or more local (Layer 2) prefixes (up to a maximum of 16), which are formatted in the style of a 4-octet IPv4 address. A local prefix identifies a T-PE or S-PE in the PW routing domain.

c. For each local prefix, at least one 8-byte route distinguisher can be configured. It is also possible to configure an optional BGP community attribute.

For each local prefix, BGP then advertises each global ID/prefix tuple and unique RD and community pseudowire using the MS-PW NLRI, based on the aggregated FEC129 AII Type 2 and the Layer 2 VPN/PW routing AFI/SAFI 25/6, to each T-PE/S-PE that is a T-LDP neighbor, subject to local BGP policies.

The dynamic advertisement of each of these pseudowire routes is enabled for each prefix and RD using the **advertise-bgp** command.

An export policy is also required in order to export MS-PW routes in MP-BGP. This can be done using a default policy, such as the following:

```
*A:lin-123>config>router>policy-options# info
----------------------------------------------
            policy-statement "ms-pw"
                default-action accept
                exit
            exit
----------------------------------------------
```

However, this would export all routes. A recommended choice is to enable filtering per-family, as follows:

```
*A:lin-123>config>router>policy-options# info
----------------------------------------------
            policy-statement "to-mspw"
                entry 1
                    from
                        family ms-pw
                    exit
                    action accept
                    exit
                exit
            exit
----------------------------------------------
```

The following command is then added in the **config>router>bgp** context.

```
        export "to-mspw"
```

Local-preference for iBGP and BGP communities can be configured under such a policy.

Static Routing

In addition to support for BGP routing, static MS-PW routes may also be configured using the **config>services>pw-routing>static-route** command. Each static route comprises the target T-PE Global-ID and prefix, and the IP address of the T-LDP session to the next hop S-PE or T-PE that should be used.

If a static route is set to 0, then this represents the default route. If a static route exists to a given T-PE, then this is used in preference to any BGP route that may exist.

Explicit Paths

A set of default explicit routes to a remote T-PE or S-PE prefix may be configured on a T-PE under **config>services>pw-routing** using the **path name** command. Explicit paths are used to populate the explicit route TLV used by MS-PW T-LDP signaling. Only strict (fully qualified) explicit paths are supported.

Note that it is possible to configure explicit paths independently of the configuration of BGP or static routing.

Configuring VLLs using Dynamic MS-PWs

One or more spoke SDPs may be configured for distributed Epipe VLL services. Dynamic MS-PWs use FEC129 (also known as the Generalized ID FEC) with Attachment Individual Identifier (AII) Type 2 to identify the pseudowire, as opposed to FEC128 (also known as the PW ID FEC) used for traditional single segment pseudowires and for pseudowire switching. FEC129 spoke SDPs are configured under the **spoke-sdp-fec** command in the CLI.

FEC129 AII Type 2 uses a Source Attachment Individual Identifier (SAII) and a Target Attachment Individual Identifier (TAII) to identify the end of a pseudowire at the T-PE. The SAII identifies the local end, while the TAII identifies the remote end. The SAII and TAII are each structured as follows:

- Global-ID — This is a 4 byte identifier that uniquely identifies an operator or the local network.
- Prefix — A 4-byte prefix, which should correspond to one of the local prefixes assigned under pw-routing.
- AC-ID — A 4-byte identifier for this end of the pseudowire. This should be locally unique within the scope of the global-id:prefix.

Active/Passive T-PE Selection

Dynamic MS-PWs use single-sided signaling procedures with double-sided configuration, a fully qualified FEC must be configured at both endpoints. That is, one T-PE (the source T-PE, ST-PE) of the MS-PW initiates signaling for the MS-PW, while the other end (the terminating T-PE, TT-PE) passively waits for the label mapping message from the far-end and only responds with a label mapping message to set up the opposite direction of the MS-PW when it receives the label mapping from the ST-PE. By default, the 7x50 will determine which T-PE is the ST-PE (the active T-PE) and which is the TT-PE (the passive T-PE) automatically, based on comparing the SAII with the TAII as unsigned integers. The T-PE with SAII>TAII assumes the active role. However, it is possible to override this behavior using the signaling {**master** | **auto**} command under the spoke-sdp-fec. If master is selected at a given T-PE, then it will assume the active role. If a T-PE is at the endpoint of a spoke SDP that is bound to an VLL SAP and single sided auto-configuration is used (see below), then that endpoint is always passive. Therefore, signaling master should only be used when it is known that the far end will assume a passive behavior.

Automatic Endpoint Configuration

Automatic endpoint configuration allows the configuration of an endpoint without specifying the TAII associated with that spoke-sdp-fec. It allows a single-sided provisioning model where an incoming label mapping message with a TAII that matches the SAII of that spoke SDP to be automatically bound to that endpoint. This is useful in scenarios where a service provider wishes to separate service configuration from the service activation phase.

Automatic endpoint configuration is supported required for Epipe VLL spoke-sdp-fec endpoints bound to a VLL SAP. It is configured using the **spoke-sdp-fec>auto-config** command, and excluding the TAII from the configuration. When auto-configuration is used, the node assumed passive behavior from a point of view of T-LDP signaling (see above). Therefore, the far-end T-PE must be configured for signaling master for that spoke-sdp-fec.

Selecting a Path for an MS-PW

Path selection for signaling occurs in the outbound direction (ST-PE to TT-PE) for an MS-PW. In the TT-PE to ST-PE direction, a label mapping message simply follows the reverse of the path already taken by the outgoing label mapping.

A node can use explicit paths, static routes, or BGP routes to select the next hop S-PE or T-PE. The order of preference used in selecting these routes is:

1. Explicit Path
2. Static route
3. BGP route

In order to use an explicit path for an MS-PW, an explicit path must have been configured in the **config>services>pw-routing>path** *path-name* context. The user must then configure the corresponding **path** *path-name* under **spoke-sdp-fec**.

If an explicit path name is not configured, then the TT-PE or S-PE will perform a longest match lookup for a route (static if it exists, and BGP if not) to the next hop S-PE or T-PE to reach the TAII.

Pseudowire routing chooses the MS-PW path in terms of the sequence of S-PEs to use to reach a given T-PE. It does not select the SDP to use on each hop, which is instead determined at signaling time. When a label mapping is sent for a given pseudowire segment, an LDP SDP will be used to reach the next-hop S-PE/T-PE if such an SDP exists. If not, and a RFC 3107 labeled BGP SDP is available, then that will be used. Otherwise, the label mapping will fail and a label release will be sent.

Pseudowire Templates

Dynamic MS-PWs support the use of the pseudowire template for specifying generic pseudowire parameters at the T-PE. The pseudowire template to use is configured in the **spoke-sdp-fec>pw-template-bind** *policy-id* context. Dynamic MS-PWs do not support the provisioned SDPs specified in the pseudowire template.

Pseudowire Redundancy

Pseudowire redundancy is supported on dynamic MS-PWs used for VLLs. It is configured in a similar manner to pseudowire redundancy on VLLs using FEC128, whereby each spoke-sdp-fec within an endpoint is configured with a unique SAII/TAII.

Figure 27 illustrates the use of pseudowire redundancy.

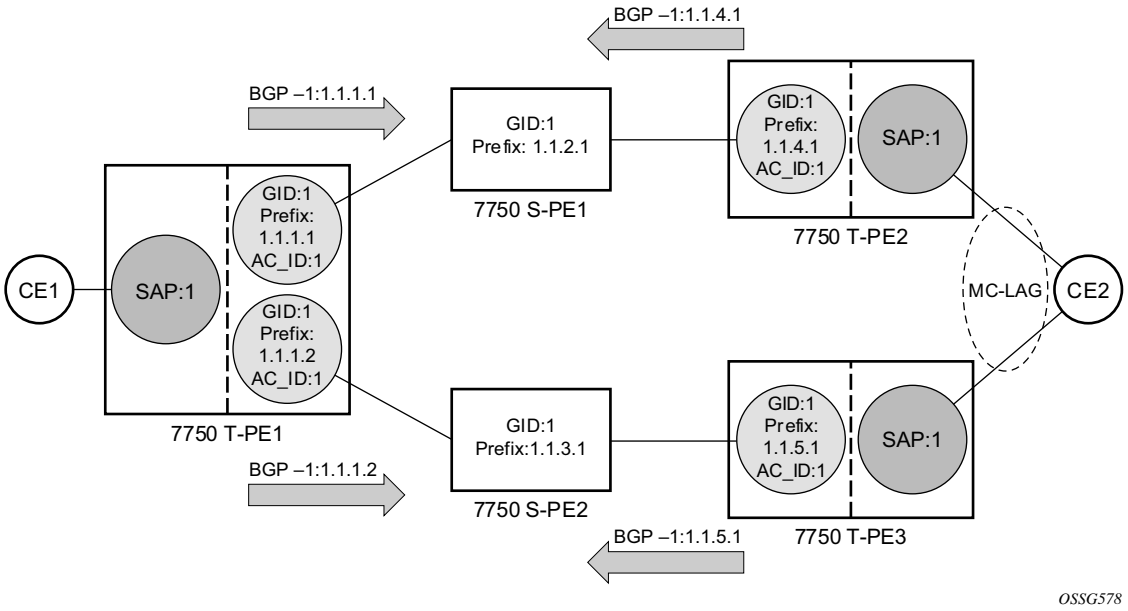


Figure 27: Pseudowire Redundancy

The following is a summary of the key points to consider in using pseudowire redundancy with dynamic MS-PWs:

- Each MS-PW in the redundant set must have a unique SAII/TAII set and is signalled separately. The primary pseudowire is configured in the **spoke-sdp-fec>primary** context.
- Each MS-PW in the redundant set should use a diverse path (from the point of view of the S-PEs traversed) from every other MS-PW in that set if path diversity is possible in a given network topology. There are a number of possible ways to achieve this:
  - → Configure an explicit path for each MS-PW.
  - → Allow BGP routing to automatically determine diverse paths using BGP policies applied to different local prefixes assigned to the primary and standby MS-PWs.
  - → Path diversity can be further provided for each primary pseudowire through the use of a BGP route distinguisher.

If the primary MS-PW fails, fail-over to a standby MS-PW, as per the normal pseudowire redundancy procedures. A configurable retry timer for the failed primary MS-PW is then started. When the timer expires, attempt to re-establish the primary MS-PW using its original path, up to a maximum number of attempts as per the retry count parameter. The T-PE may then optionally revert back to the primary MS-PW on successful reestablishment.

Note that since the SDP ID is determined dynamically at signaling time, it cannot be used as a tie breaker to choose the primary MS-PW between multiple MS-PWs of the same precedence. The user should therefore explicitly configure the precedence values to determine which MS-PW is active in the final selection.

VCCV OAM for Dynamic MS-PWs

The primary difference between dynamic MS-PWs and those using FEC128 is support for FEC129 AII type 2. As in PW Switching, VCCV on dynamic MS-PWs requires the use of the VCCV control word on the pseudowire. Both the vccv-ping and vccv-trace commands support dynamic MS-PWs.

VCCV-Ping on Dynamic MS-PWs

VCCV-ping supports the use of FEC129 AII type 2 in the target FEC stack of the ping echo request message. The FEC to use in the echo request message is derived in one of two ways: Either the user can specify only the *spoke-sdp-fec-id* of the MS-PW in the **vccv-ping** command, or the user can explicitly specify the SAII and TAII to use.

If the SAII:TAII is entered by the user in the vccv-ping command, then those values are be used for the vccv-ping echo request, but their order is be reversed before being sent so that they match the order for the downstream FEC element for an S-PE, or the locally configured SAII:TAII for a remote T-PE of that MS-PW. Note that is SAII:TAII is entered in addition to the *spoke-sdp-fec-id*, then the system will verify the entered values against the values stored in the context for that *spoke-sdp-fec-id*.

Otherwise, if the SAII:TAII to use in the target FEC stack of the vccv-ping message is not entered by the user, and if a switching point TLV was previously received in the initial label mapping message for the reverse direction of the MS-PW (with respect to the sending PE), then the SAII:TAII to use in the target FEC stack of the vccv-ping echo request message is derived by parsing that switching point TLV based on the user-specified TTL (or a TTL of 255 if none is specified). In this case, the order of the SAII:TAII in the switching point TLV is maintained for the vccv-ping echo request message.

If no pseudowire switching point TLV was received, then the SAII:TAII values to use for the vccv-ping echo request are derived from the MS-PW context, but their order is reversed before being sent so that they match the order for the downstream FEC element for an S-PE, or the locally configured SAII:TAII for a remote T-PE of that MS-PW.

Note that the use of *spoke-sdp-fec-id* in vccv-ping is only applicable at T-PE nodes, since it is not configured for a given MS-PW at S-PE nodes.

VCCV-Trace on Dynamic MS-PWs

The 7x50 supports the MS-PW path trace mode of operation for VCCV trace, as per pseudowire switching, but using FEC129 AII type 2. As in the case of vccv-ping, the SAII:TAII used in the VCCV echo request message sent from the T-PE or S-PE from which the VCCV trace command is executed is specified by the user or derived from the context of the MS-PW. Note that the use of *spoke-sdp-fec-id* in vccv-trace is only applicable at T-PE nodes, since it is not configured for a given MS-PW at S-PE nodes.

Example Dynamic MS-PW Configuration

This section presents an example of how to configure Dynamic MS-PWs for a VLL service between a set of 7x50 nodes. The network consists of two 7x50 T-PEs and two 7x50 playing the role of S-PEs, as shown in the following figure. Each 7x50 peers with its neighbor using LDP and BGP.

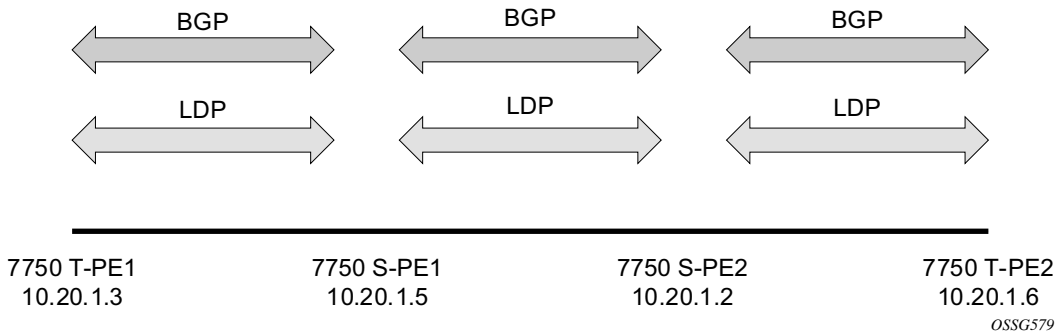


Figure 28: Dynamic MS-PW Example

The example uses BGP to route dynamic MS-PWs and T-LDP to signal them. Therefore each node must be configured to support the MS-PW address family under BGP, and BGP and LDP peerings must be established between the T-PEs/S-PEs. The appropriate BGP export policies must also be configured.

Next, pseudowire routing must be configured on each node. This includes an S-PE address for every participating node, and one or more local prefixes on the T-PEs. MS-PW paths and static routes may also be configured.

Once this routing and signaling infrastructure is established, spoke-sdp-fecs can be configured on each of the T-PEs.

T-PE-1

```
config
   router
      ldp
         targeted-session
            peer 10.20.1.5
            exit
         exit
      policy-options
         begin
         policy-statement "exportMsPw"
            entry 10
               from
                  family ms-pw
               exit
               action accept
               exit
            exit
         exit
         commit
      exit
      bgp
         family ms-pw
         connect-retry 1
         min-route-advertisement 1
         export "exportMsPw"
         rapid-withdrawal
         group "ebgp"
            neighbor 10.20.1.5
               multihop 255
               peer-as 200
            exit
         exit
     exit
config
   service
      pw-routing
         spe-address 3:10.20.1.3
         local-prefix 3:10.20.1.3 create
         exit
         path "path1_to_F" create
            hop 1 10.20.1.5
            hop 2 10.20.1.2
            no shutdown
         exit
      exit
      epipe 1 customer 1 vpn 1 create
         description "Default epipe
              description for service id 1"
         service-mtu 1400
         service-name "XYZ Epipe 1"
         sap 2/1/1:1 create
         exit
         spoke-sdp-fec 1 fec 129 aii-type 2 create
            retry-timer 10
            retry-count 10
            saii-type2 3:10.20.1.3:1
            taii-type2 6:10.20.1.6:1
            no shutdown
         exit
         no shutdown
      exit
```

T-PE-2

```
   config
   router
      ldp
         targeted-session
            peer 10.20.1.2
            exit
         exit
         …
      policy-options
         begin
         policy-statement "exportMsPw"
            entry 10
               from
                  family ms-pw
               exit
               action accept
               exit
            exit
         exit
         commit
      exit

      bgp
         family ms-pw
         connect-retry 1
         min-route-advertisement 1
         export "exportMsPw"
         rapid-withdrawal
         group "ebgp"
            neighbor 10.20.1.2
               multihop 255
               peer-as 300
            exit
         exit
      exit
config
   service
      pw-routing
         spe-address 6:10.20.1.6
         local-prefix 6:10.20.1.6 create
         exit
         path "path1_to_F" create
            hop 1 10.20.1.2
            hop 2 10.20.1.5
            no shutdown
         exit
      exit
      epipe 1 customer 1 vpn 1 create
         description "Default epipe
              description for service id 1"
service-mtu 1400
         service-name "XYZ Epipe 1"
         sap 1/1/3:1 create
         exit
         spoke-sdp-fec 1 fec 129 aii-type 2 create
            retry-timer 10
            retry-count 10
            saii-type2 6:10.20.1.6:1
            taii-type2 3:10.20.1.3:1
            no shutdown
         exit
         no shutdown
```

S-PE-1

```
config
   router
      ldp
         targeted-session
            peer 10.20.1.3
            exit
            peer 10.20.1.2
            exit
         exit
         …
      bgp
         family ms-pw
         connect-retry 1
         min-route-advertisement 1
         rapid-withdrawal
         group "ebgp"
            neighbor 10.20.1.2
               multihop 255
               peer-as 300
            exit
            neighbor 10.20.1.3
               multihop 255
               peer-as 100
            exit
         exit
     exit

   service
      pw-routing
         spe-address 5:10.20.1.5
      exit
```

S-PE-2

```
config
   router
      ldp
         targeted-session
            peer 10.20.1.5
            exit
            peer 10.20.1.6
            exit
         exit
         …
      bgp
         family ms-pw
         connect-retry 1
         min-route-advertisement 1
         rapid-withdrawal
         group "ebgp"
            neighbor 10.20.1.5
               multihop 255
               peer-as 200
            exit
            neighbor 10.20.1.6
               multihop 255
               peer-as 400
            exit
         exit
     exit

service
      pw-routing
         spe-address 2:10.20.1.2
      exit
```

VLL Resilience with Two Destination PE Nodes

Figure 29 illustrates the application of pseudowire redundancy to provide Ethernet VLL service resilience for broadband service subscribers accessing the broadband service on the service provider BRAS.

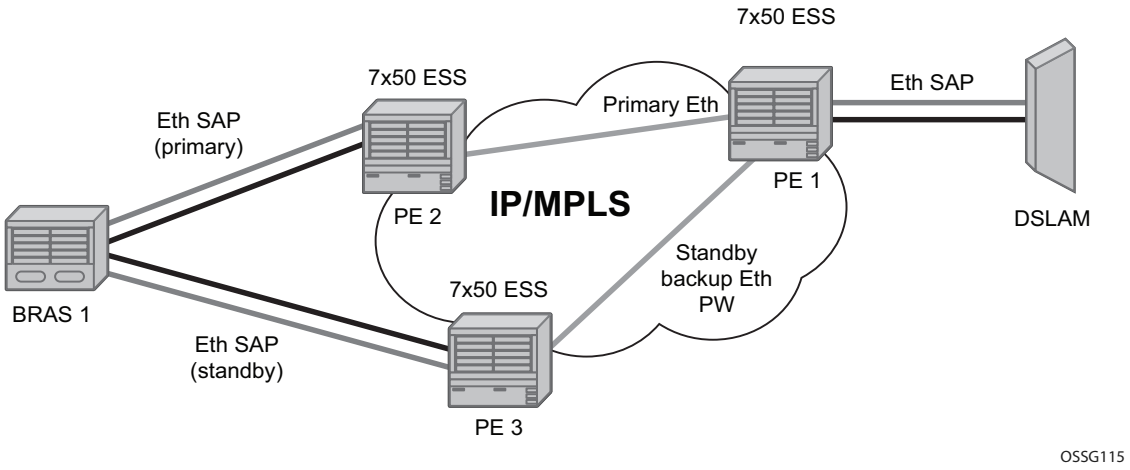


Figure 29: VLL Resilience

If the Ethernet SAP on PE2 fails, PE2 notifies PE1 of the failure by either withdrawing the primary pseudowire label it advertised or by sending a pseudowire status notification with the code set to indicate a SAP defect. PE1 will receive it and will immediately switch its local SAP to forward over the secondary standby spoke SDP. In order to avoid black holing of in-flight packets during the switching of the path, PE1 will accept packets received from PE2 on the primary pseudowire while transmitting over the backup pseudowire. However, in other applications such as those described in Access Node Resilience Using MC-LAG and Pseudowire Redundancy on page 141, it will be important to minimize service outage to end users.

When the SAP at PE2 is restored, PE2 updates the new status of the SAP by sending a new label mapping message for the same pseudowire FEC or by sending pseudowire status notification message indicating that the SAP is back up. PE1 then starts a timer and reverts back to the primary at the expiry of the timer. By default, the timer is set to 0, which means PE1 reverts immediately. A special value of the timer (infinity) will mean that PE1 should never revert back to the primary pseudowire.

The behavior of the pseudowire redundancy feature is the same if PE1 detects or is notified of a network failure that brought the spoke SDP operational status to DOWN. The following are the events which will cause PE1 to trigger a switchover to the secondary standby pseudowire:

1. T-LDP peer (remote PE) node withdrew the pseudowire label.

2. T-LDP peer signaled a FEC status indicating a pseudowire failure or a remote SAP failure.
3. T-LDP session to peer node times out.
4. SDP binding and VLL service went down as a result of network failure condition such as the SDP to peer node going operationally down.

The SDP type for the primary and secondary pseudowires need not be the same. In other words, the user can protect a RSVP-TE based spoke SDP with a LDP or GRE based one. This provides the ability to route the path of the two pseudowires over different areas of the network. All VLL service types, for example, Apipe, Epipe, Fpipe, and Ipipe are supported.

Alcatel-Lucent's routers support the ability to configure multiple secondary standby pseudowire paths. For example, PE1 uses the value of the user configurable precedence parameter associated with each spoke SDP to select the next available pseudowire path after the failure of the current active pseudowire (whether it is the primary or one of the secondary pseudowires). The revertive operation always switches the path of the VLL back to the primary pseudowire though. There is no revertive operation between secondary paths meaning that the path of the VLL will not be switched back to a secondary pseudowire of higher precedence when the latter comes back up again.

Alcatel-Lucent's routers support the ability for a user-initiated manual switchover of the VLL path to the primary or any of the secondary be supported to divert user traffic in case of a planned outage such as in node upgrade procedures.

This application can make use of all types of VLL supported on SR-Series routers. However, if a SAP is configured on a MC-LAG instance, only the Epipe service type is allowed.

Master-Slave Operation

Master-Slave pseudowire redundancy is discussed in this section. It adds the ability for the remote peer to react to the pseudowire standby status notification, even if only one spoke-SDP terminates on the VLL endpoint on the remote peer, by blocking the transmit (Tx) direction of a VLL spoke SDP when the far-end PE signals standby. This solution enables the blocking of the Tx direction of a VLL spoke SDP at both master and slave endpoints when standby is signalled by the master endpoint. This approach satisfies a majority of deployments where bidirectional blocking of the forwarding on a standby spoke SDP is required.

Figure 30 illustrates the operation of master-slave pseudowire redundancy. In this scenario, an Epipe service is provided between CE1 and CE2. CE2 is dual homed to PE2 and PE3, and thus PE1 is dual-homed to PE2 and PE3 using Epipe spoke SDPs. The objectives of this feature is to ensure that only one pseudowire is used for forwarding in both directions by PE1, PE2 and PE3 in the absence of a native dual homing protocol between CE2 and PE2/PE3, such as MC-LAG. In normal operating conditions (the SAPs on PE2 and PE3 towards CE2 are both up and there are no defects on the ACs to CE2), PE2 and PE3 cannot choose which spoke SDP to forward on based on the status of the AC redundancy protocol.

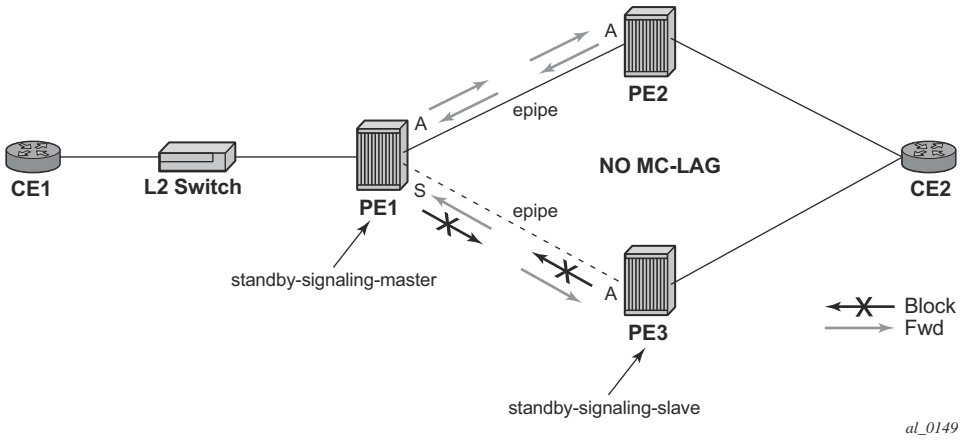


Figure 30: Master-Slave Pseudowire Redundancy

Master-slave pseudowire redundancy adds the ability for the remote peer to react to the pseudowire standby status notification, even if only one spoke SDP terminates on the VLL endpoint on the remote peer. When the CLI command **standby-signaling-slave** is enabled at the spoke SDP or explicit endpoint level in PE2 and PE3, then any spoke SDP for which the remote peer signals PW FWD Standby will be blocked in the transmit direction.

This is achieved as follows. The **standby-signaling-master** state is activated on the VLL endpoint in PE1. In this case, a spoke SDP is blocked in the transmit direction at this master endpoint if it is either in operDown state, or it has lower precedence than the highest precedence spoke SDP, or the given peer PE signals one of the following pseudowire status bits:

- Pseudowire not forwarding (0x01)
- SAP (ingress) receive fault (0x02)
- SAP (egress) transmit fault (0x04)
- SDP binding (ingress) receive fault (0x08)
- SDP binding (egress) transmit fault (0x10)

The fact that the given spoke SDP has been blocked will be signaled to LDP peer through the pseudowire status bit (PW FWD Standby (0x20)). This will prevent traffic being sent over this spoke SDP by the remote peer, but obviously only in case that remote peer supports and reacts to pseudowire status notification. Previously, this applied only if the spoke SDP terminates on an IES, VPRN or VPLS. However, if standby-signaling-slave is enabled at the remote VLL endpoint then the Tx direction of the spoke SDP will also be blocked, according to the rules in Operation of Master-Slave Pseudowire Redundancy with Existing Scenarios on page 119.

Note that although master-slave operation provides bidirectional blocking of a standby spoke SDP during steady-state conditions, it is possible that the Tx directions of more than one slave endpoint can be active for transient periods during a fail-over operation. This is due to slave endpoints

transitioning a spoke SDP from standby to active receiving and/or processing a pseudowire preferential forwarding status message before those transitioning a spoke SDP to standby. This transient condition is most likely when a forced switch-over is performed, or the relative preferences of the spoke SDPs is changed, or the active spoke SDP is shutdown at the master endpoint. During this period, loops of unknown traffic may be observed. Fail-overs due to common network faults that can occur during normal operation, a failure of connectivity on the path of the spoke SDP or the SAP, would not result in such loops in the data path.

Interaction with SAP-Specific OAM

If all of the spoke SDPs bound to a SAP at a slave PE are selected as standby, then this should be treated from a SAP OAM perspective in the same manner as a fault on the service, an SDP-binding down or remote SAP down. That is, a fault should be indicated to the service manager. If SAP-specific OAM is enabled towards the CE, such as Ethernet CCM, E-LMI, or FR LMI, then this should result in the appropriate OAM message being sent on the SAP. This can enable the remote CE to avoid forwarding traffic towards a SAP which will drop it.

Figure 31 shows an example for the case of Ethernet LMI.

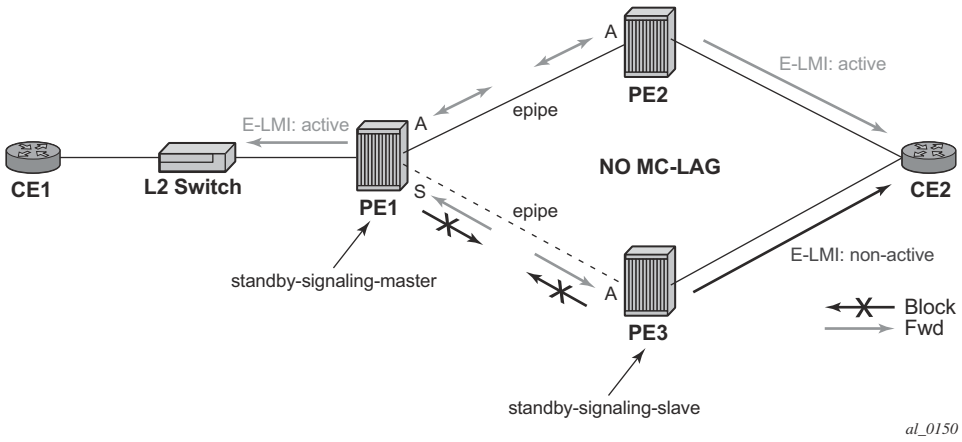


Figure 31: Example of SAP OAM Interaction with Master-Slave Pseudowire Redundancy

Local Rules at Slave VLL PE

It is not possible to configure a standby-signaling-slave on endpoints or spoke SDPs bound to an IES, VPRN, ICB, MC-EP or that form part of an MC-LAG or MC-APS.

If **standby-signaling-slave** is configured on a given spoke SDP or explicit endpoint, then the following rules apply. Note that the rules describe the case of several spoke SDPs in an explicit endpoint. The same rules apply to the case of a single spoke SDP outside of an endpoint where no endpoint exists:

- Rules for processing endpoint SAP active/standby status bits:
  - → Since the SAP in endpoint X is never a part of a MC-LAG/MC-APS instance, a forwarding status of ACTIVE is always advertised.
- Rules for processing and merging local and received endpoint object status Up/Down operational status:

1. Endpoint ‘X’ is operationally UP if at least one of its objects is operationally UP. It is Down if all its objects are operationally down.
2. If all objects in endpoint ‘X’ transition locally to Down state, and/or received a “SAP Down” notification via remote T-LDP status bits or via SAP specific OAM signal, and/or received status bits of “SDP-binding down”, and/or received status bits of “PW not forwarding”, the node must send status bits of “SAP Down” over all ‘Y’ endpoint spoke SDPs.
3. Endpoint ‘Y’ is operationally UP if at least one of its objects is operationally UP. It is Down if all its objects are operationally down.
4. If a spoke SDP in endpoint ‘Y’, including the ICB spoke SDP, transitions locally to Down state, the node must send T-LDP “SDP-binding down” status bits on this spoke SDP.
5. If a spoke SDP in endpoint ‘Y’, received T-LDP “SAP down” status bits, and/or received T-LDP “SDP-binding down” status bits, and/or received status bits of “PW not forwarding”, the node saves this status and takes no further action. The saved status is used for selecting the active transmit endpoint object as per the pseudo-code in Section 5.1.2.
6. If, all objects in endpoint ‘Y’, or a single spoke SDP that exists outside of an endpoint (and no endpoint exists), transition locally to down state, and/or received T-LDP “SAP Down” status bits, and/or received T-LDP “SDP-binding down” status bits, and/or received status bits of “PW not forwarding”, and/or the received status bits of ‘PW FWD standby’, the node must send a “SAP down” notification on the ‘X’ endpoint SAP via the SAP specific OAM signal, if applicable.
7. If the peer PE for a given object in endpoint ‘Y’ signals ‘PW FWD standby’, the spoke SDP must be blocked in the transmit direction and the spoke SDP is not eligible for selection by the active transmit selection rules.
8. If the peer PE for a given object in endpoint ‘Y’ does not signal ‘PW FWD standby’, then spoke SDP is eligible for selection.

Operation of Master-Slave Pseudowire Redundancy with Existing Scenarios

This section discusses how master-slave pseudowire redundancy could operate.

VLL Resilience

Figure 32 displays a VLL resilience path example. An sample configuration follows.

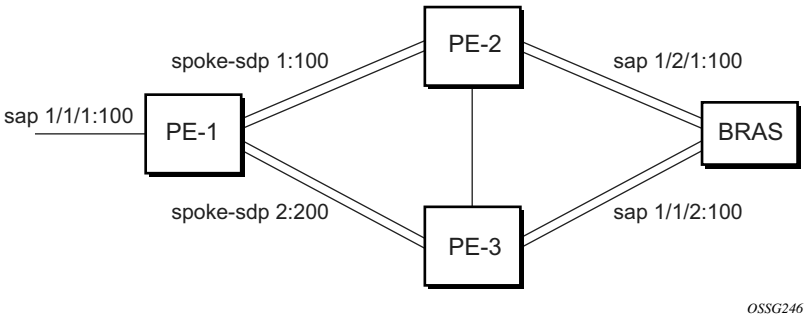


Figure 32: VLL Resilience

Note that a **revert-time** value of zero (default) means that the VLL path will be switched back to the primary immediately after it comes back up

```
PE1
configure service epipe 1
     endpoint X
     exit
     endpoint Y
     revert-time 0
     standby-signaling-master
     exit
     sap 1/1/1:100 endpoint X
     spoke-sdp 1:100 endpoint Y
precedence primary
     spoke-sdp 2:200 endpoint Y
precedence 1
PE2
configure service epipe 1
     endpoint X
     exit
     sap 2/2/2:200 endpoint X
     spoke-sdp 1:100
          standby-signaling-slave
```

PE3

```
configure service epipe 1
     endpoint X
     exit
     sap 3/3/3:300 endpoint X
     spoke-sdp 2:200
          standby-signaling-slave
```

VLL Resilience for a Switched Pseudowire Path

Figure 33 displays a VLL resilience for a switched pseudowire path example. A sample configuration follows.

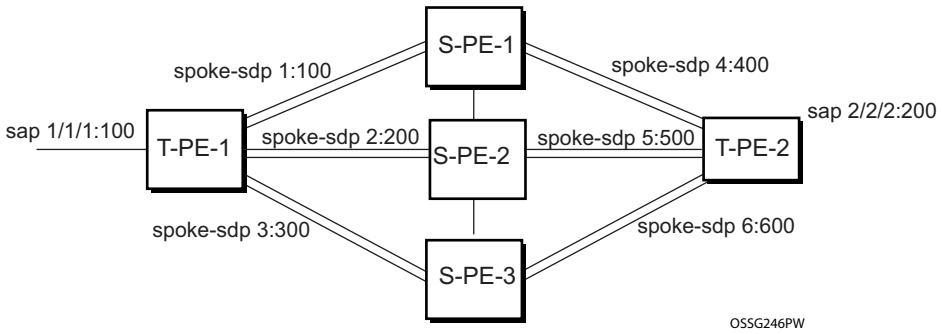


Figure 33: VLL Resilience with Pseudowire Switching

Configuration

```
T-PE1
configure service epipe 1
     endpoint X
     exit
     endpoint Y
     revert-time 100
     standby-signaling-master
     exit
     sap 1/1/1:100 endpoint X
     spoke-sdp 1:100 endpoint Y
          precedence primary
     spoke-sdp 2:200 endpoint Y
          precedence 1
     spoke-sdp 3:300 endpoint Y
          precedence 1

T-PE2
configure service epipe 1
     endpoint X
     exit
     endpoint Y
     revert-time 100
     standby-signaling-slave
     exit
     sap 2/2/2:200 endpoint X
     spoke-sdp 4:400 endpoint Y
```

```
        precedence primary
    spoke-sdp 5:500 endpoint Y
        precedence 1
    spoke-sdp 6:600 endpoint Y
        precedence 1
```

S-PE1

VC switching indicates a VC cross-connect so that the service manager does not signal the VC label mapping immediately but will put this into passive mode.

```
configure service epipe 1 vc-switching
    spoke-sdp 1:100
    spoke-sdp 4:400
```

Pseudowire SAPs

Refer to the *SR OS Layer 3 Services Guide* for details of how to use pseudowire SAPs with Layer-2 services.

Epipe Using BGP-MH Site Support for Ethernet Tunnels

Using Epipe in combination with G.8031 and BGP Multi-Homing in the same manner as VPLS offers a multi-chassis resiliency option for Epipe services that is a non-learning and non-flooded service. Note that MC-LAG (see, Access Node Resilience Using MC-LAG and Pseudowire Redundancy on page 141) offers access node redundancy with active/stand-by links while Ethernet Tunnels offers per service redundancy with all active links and active or standby services. G.8031 offers an end to end service resiliency for Epipe and VPLS services. BGP-MH Site Support for Ethernet Tunnels offers Ethernet edge resiliency for Epipe services that integrates with MPLS Pseudowire Redundancy.

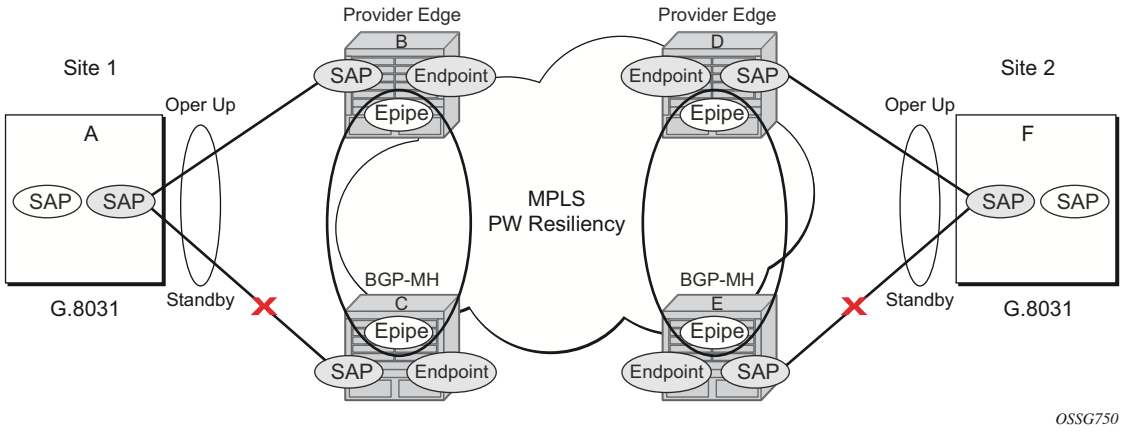


Figure 34: BGP-MH Site Support for Ethernet Tunnels

Figure 34 shows the BGP-MH Site Support for Ethernet Tunnels; where a G.8031 edge device (A) is configure to two provider edge switches (B and C). G.8031 is configured on the Access devices (A and F). An Epipe Endpoint service is configured along with BGP Multi-homing and Pseudowire Redundancy on the provider edge nodes (B,C and D,E). This configuration offers a fully redundant Epipe service.

Operational Overview

G.8031offers a number of redundant configurations. Normally it offers the ability to control two independent paths for 1:1 protection. In the BGP-MH Site Support for Ethernet Tunnels case, BGP drives G.8031 as a slave service. In this case, the Provider Edge operates using only standard 802.1ag MEPs with CCM to monitor the paths. Figure 35 shows an Epipe service on a Customer Edge (CE) device that uses G.8031 with two paths and two MEPs. The Paths can use a single VLAN of DOT1Q or QinQ encapsulation.

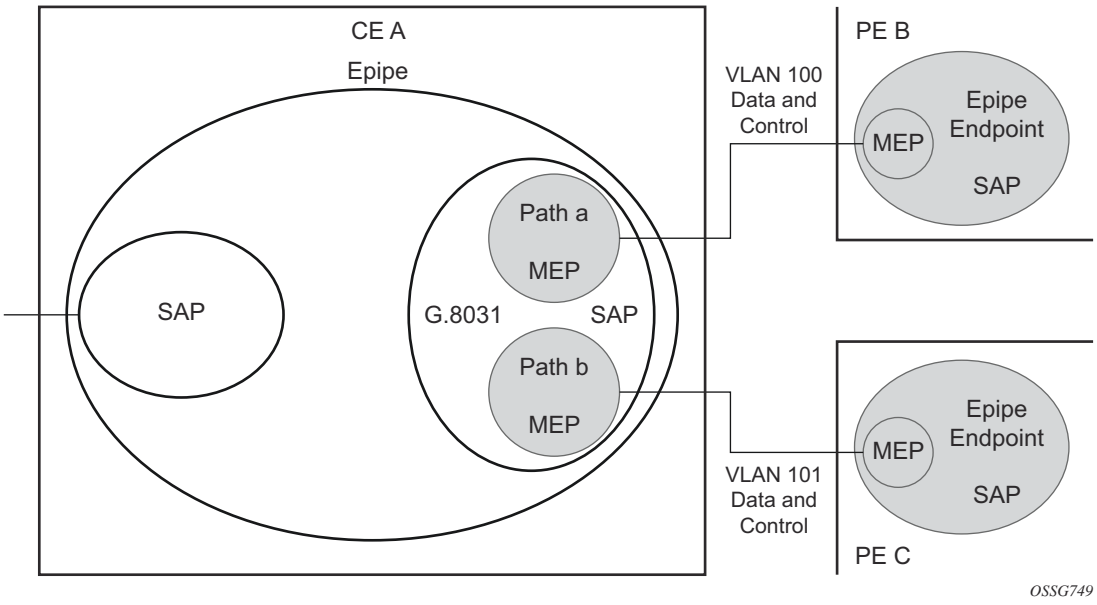


Figure 35: G.8031 for Slave Operation

In a single-service deployment the control (CFM) and data will share the same port and VID. For multiple services for scaling fate sharing is allowed between multiple SAPs, but all SAPs within a group must be on the same physical port.

To get fate sharing for multiple services with this feature, a dedicated G.8031 CE based service (one VLAN) is connect to a Epipe SAP on a PE which uses BGP-MH and operational groups to control other G.8031 tunnels. This dedicated G.8031 still has a data control capabilities, but the data Epipe service is not bearing user data packets. On the CE, this dedicated G.8031 is only used for group control. The choice of making this a dedicated Control for a set of G.8031 tunnels is merely to simplify operation and allow individual disabling of services. Using a dedicated G.8031 for both control and to carry data traffic is allowed.

Fate sharing from the PE side is achieved using BGP and operational groups. G.8031 Epipe services can be configured on the CE as regular non fate shared G.8031 services but due to the configuration on the PE side, these Ethernet Tunnels will be treated as a group following the one designated control service. The G.8031 control logic on the CE is slaved to the BGP-MH control.

On the CE G.8031 allows independent configuration of VIDs on each path. On the PE the Epipe or Endpoint that connects to the G.8031 must have a SAP with the corresponding VID. If the G.8031 service has a Maintenance End Point (MEP) for that VID, the SAP should be configured with a MEP. The MEPs on the paths on the CE signal standard interface status TLV (ifStatusTLV), No Fault (Up) and Fault (Down). The MEPs on the PE (Epipe or Endpoint) also use signaling of ifStatusTlv No Fault, and Fault to control the G.8031 SAP. However in the 7x50 model fate shared Ethernet Tunnels with no MEP are allowed. In this case it is up to the CE to manage these CE based fate shared tunnels.

Interfaces status signaling (ifStatusTLV) is used to control the G.8031 tunnel form the PE side. Normally the CE will signal No Fault in the path SAP MEP inStatusTLV before the BGP-MH will cause the SAP MEP to become active by signaling No Fault.

Detailed Operation

For this feature, BGP-MH is used the master control and the Ethernet Tunnel is a slave. The G.8031 on the CE is unaware that it is being controlled. While a single Epipe service is configured and will serve as the control for the CE connection allowing fate sharing all signaling to the CE is based on the ifstatusTLV per G.8031 tunnel. Note with G.8031 by controlling it with BGP-MH, the G.8031 CE is forced to be slaved to the PE BGP-MH election.BGP-MH election is control by the received VPLS preference or BGP local-preference or PE Id (IP address of Provider Edge) if local-preference is equal. There may be traps generated on the CE side for some G.8031 implementations but these can be suppressed or filtered to allow this feature to operate.

There are two configuration options:

- Every G.8031 service SAP terminates on a single Epipe that has BGP-MH. These Epipes may utilize endpoints with or without ICBs.
- A control Epipe service that monitors a single SAP that is used for group control of fate shared CE services. In this case, the Epipe service has a SAP that serves as the control termination for one Ethernet Tunnel connection. The group fate sharing SAPs may or may not have MEPs if they use shared fate. In this case the Epipe may have endpoints but will not support ICBs.

The MEP ifStatusTlv and CCM are used for monitoring the PE to CE SAP. MEP ifStatusTlv is used to signal, the Ethernet Tunnel inactive and is used CCM as an aliveness mechanism. There is no G.8031 logic on the PE, the SAP is simply controlling the correspond CE SAP.

Sample Operation of G.8031 BGP-MH

Any Ethernet tunnel actions (force, lock) on the CE (single site) do not control the action to switch paths directly but they may influence the outcome of BGP-MH if they are on a control tunnel. If a path is disabled on the CE the result may force the SAP with an MEP on the PE to eventually take the SAP down but it is suggested to run commands from the BGP-MH side to control these connections.

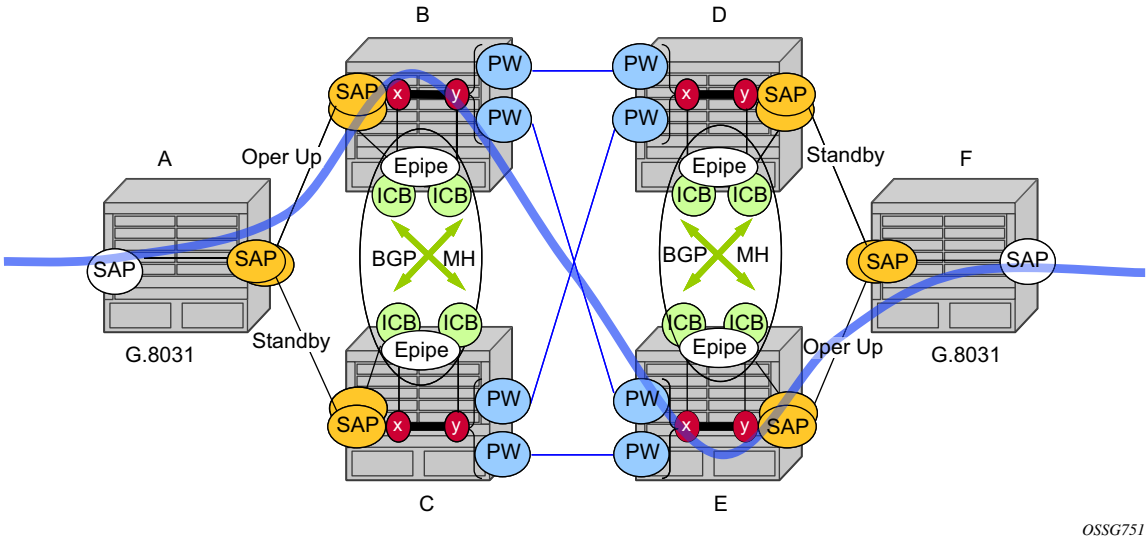


Figure 36: Full Redundancy G.8031 Epipe & BGP-MH

Table 9 lists the SAP MEP signaling shown in Figure 36. For a description of the events shown in this sample operation, see Events in Sample Operation on page 127.

Table 9: SAP MEP Signaling

	G.8031 ET on CE	Path A MEP Facing Node B Local ifStatus	Path B MEP Facing Node C Local ifStatus	Path B PE MEP ifStatus	Path B PE MEP ifStatus
1	Down (inactive)	No Fault[a]	No Fault	Fault	Fault
2	Up use Path A	No Fault	No Fault	No Fault	Fault
3	Up use Path B	No Fault	No Fault	Fault	No Fault
4	Down Path a fault	Fault[b]	No Fault	Fault	Fault
5	Down Path A & B fault at A	Fault	No Fault	Fault	Fault
6	Partitioned Network Use Path Precedence Up use Path A	No Fault	No Fault	No Fault	No Fault

a. No Fault = no ifStatusTlv transmit | CCM transmit normally
b. Fault = ifStatusTlv transmit down | no CCM transmit

Events in Sample Operation

The following represents a walk through of the events for switchover in Figure 36. This configuration uses operational groups. The nodes of interest are A, B and C listed in Table 9.

1. A single G.8031 SAP that represents the control for a group of G.8031 SAPs is configured on the CE.
   - → The Control SAP does not normally carry any data, however it can if desired.
   - → An Epipe service is provisioned on each PE node (B,C) purely for control (no customer traffic flows over this service).
   - → On CE A, there is an Epipe Ethernet Tunnel (G.8031) control SAP.
   - → The Ethernet Tunnel has two paths:
     - one facing B
     - one facing C.
   - → PE B has an Epipe control SAP that is controlled by BGP-MH site and PE C also has the corresponding SAP that is controlled by the same BGP-MH site.

2. At node A, there are MEPs configured under each path that check connectivity on the A-B and A-C links. At nodes B and C, there is a MEP configured under their respective SAPs with fault propagation enabled with use ifStatusTlv.
3. Initially, assume there is no link failure:
   - → SAPs on node A have ifStatusTlv No Fault to B and C (no MEP fault detected at A); see Table 9 row 1 (Fault is signaled in the other direction PE to CE).
   - → BGP-MH makes its determination of the master or Designated Forwarder (DF).
   - → Assume SAP on node B is picked as the DF.
   - → The MEP at Path A-B signals ifStatusTlv No Fault. Due to this signal, the MEP under the node A path facing node B, detects the path to node B is usable by the path manager on A.
4. At the CE node A, Path A-C becomes standby and is brought down; see Table 9 row 2.
   - → Since fault propagation is enabled under the SAP node C MEP, and ifStatusTlv is operationally Down is remains in the present state.
   - → Under these conditions, the MEP under the node A path facing node C detects the fault and informs Ethernet manager on node A.
   - → Node A then considers bringing path A-C down.
   - → ET port remains up since path A-B is operationally up. This is a stable state.
5. On nodes B and C, each Epipe controlled SAP is the sole (controlling) member of an operational-group.
   - → Other data SAPs may be configured for fate shared VLANs (Ethernet Tunnels) and to monitor the control SAP.
   - → The SAPs facing the CE node A share the fate of the control SAP and follow the operation.
6. If there is a break in path A-B connectivity (CCM time out or LOS on the port for link A-B), then on node A the path MEP detects connectivity failure and informs Ethernet Tunnel Manger; see Table 9 row 4.
7. At this point the Ethernet Tunnel is down since both path A-B and path A-C are down.
8. The CE node A Ethernet Tunnel goes down.
9. Node B on the PE the SAP also detects the failure and the propagation of fault status goes to BGP-MH; see Table 9 row 4.
10. This in turn feeds into BGP-MH which deems the site non-DF and makes the site standby.
11. Since the SAP at Node B is standby, Service Manager feeds this to CFM, which then propagates a Fault towards Node A. This is a cyclic fault propagation. However, since path A-B is broken, the situation is stable; see Table 9 row 5.
12. There is traffic loss during the BGP-MH convergence.
    - → Load sharing mode is recommended when using a 7450 as a CE node A device.
    - → BGP-MH signals that node C is now the DF; see Table 9 row 3.

13. BGP-MH on node C elects sap and bring it up.
14. ET port transitions to port A-C is operationally up. This is a stable state. The A-C SAPs monitoring the operational-group on C transitions to operationally up.

Unidirectional failures: At point 6 the failure was detected at both ends. In the case of a unidirectional failure, CCM times out on one side.

1. In the case where the PE detects the failure, it propagates the failure to BGP-MH and the BGP-MH takes the site down causing the SAPs on the PE to signal to the CE Fault.
2. In the case of G.8031 on the CE detecting the failure, it takes the tunnel down and signals a fault to the PE, and then the SAP propagates that to BGP-MH.

BGP-MH Site Support for Ethernet Tunnels Operational-Group Model

For operational groups, one or more services follow the controlling service. On node A, there is an ET SAP facing nodes B/C, and on nodes B/C there are SAPs of the Epipe on physical ports facing node A. Each of the PE data SAPs monitor their respective operational groups, meaning they are operationally up, or down based on the operational status of the control SAPs. On node A, since the data SAP is on the ET logical port, it goes operationally down whenever the ET port goes down and similarly for going operationally up.

Alternatively, an Epipe Service may be provisioned on each node for each G.8031 data SAP (one for one service with no fate sharing). On CE node A, there will be a G.8031 Ethernet Tunnel. The Ethernet Tunnel has two paths: one facing node B and one facing node C. This option is the same as the control SAP, but there are no operational groups. However, now there is a BGP-MH Site per service. For large sites operational groups are more efficient.

BGP-MH Specifics for MH Site Support for Ethernet Tunnels

BGP Multi-Homing for VPLS on page 443 describes the procedures for using BGP to control resiliency for VPLS. These procedures are the same except that an Epipe service can be configured for BGP-MH.

PW Redundancy for BGP MH Site Support for Ethernet Tunnels

Pseudowire Redundancy Service Models on page 145 and Figure 39 on page 143 are used for the MPLS network resiliency. BGP MH Site Support for Ethernet Tunnels reuses this model.

T-LDP Status Notification Handling Rules of BGP-MH Epipes

Using Figure 39 as a reference, the following are the rules for generating, processing, and merging T-LDP status notifications in VLL service with endpoints.

Rules for Processing Endpoint SAP Active/Standby Status Bits

1. The advertised admin forwarding status of Active/Standby reflects the status of the local Epipe SAP in BGP-MH instance. If the SAP is not part of a MC-LAG instance or a BGP-MH instance, the forwarding status of Active is always advertised.
2. When the SAP in endpoint X is part of a BGP-MH instance, a node must send T-LDP forwarding status bit of SAP Active/Standby over all Y endpoint spoke-SDPs, except the ICB spoke-SDP whenever this (BGP-MH designated forwarder) status changes. The status bit sent over the ICB is always zero (Active by default).
3. When the SAP in endpoint X is not part of a MC-LAG instance or BGP-MH instance, then the forwarding status sent over all Y endpoint spoke-SDPs should always be set to zero (Active by default).
4. The received SAP Active/Standby status is saved and used for selecting the active transmit endpoint object Pseudowire Redundancy procedures.

Rules for Processing, Merging Local, and Received Endpoint Operational Status

1. Endpoint X is operationally Up if at least one of its objects is operationally Up. It is Down if all its objects are operationally Down.
2. If the SAP in endpoint X transitions locally to the Down state, or received a *SAP Down* notification via SAP specific OAM signal (SAP MEP), the node must send T-LDP *SAP Down* status bits on the Y endpoint ICB spoke-SDP only. BGP-MH SAP support MEPs for ifStatusTlv signaling. All other SAP types cannot exist on the same endpoint as an ICB spoke-SDP since non Ethernet SAP cannot be part of a MC-LAG instance or a BGP-MH Instance.
3. If the ICB spoke-SDP in endpoint X transitions locally to Down state, the node must send T-LDP *SDP-binding Down* status bits on this spoke-SDP.
4. If the ICB spoke-SDP in endpoint X received T-LDP *SDP-binding Down* status bits or *PW not forwarding* status bits, the node saves this status and takes no further action. The saved status is used for selecting the active transmit endpoint object as per the pseudo-code per Pseudowire Redundancy procedures.
5. If all objects in endpoint X transition locally to Down state due to operator or BGP-MH DF election, or received a SAP Down notification via remote T-LDP status bits or via SAP specific OAM signal (SAP MEP), or received status bits of SDP-binding Down, or received sta-

tus bits of PW not forwarding, the node must send status bits of SAP Down over all Y endpoint spoke-SDPs, including the ICB.

6. Endpoint Y is operationally Up if at least one of its objects is operationally Up. It is Down if all its objects are operationally Down.
7. If a spoke-SDP in endpoint Y, including the ICB spoke-SDP, transitions locally to Down state, the node must send T-LDP SDP-binding Down status bits on this spoke-SDP.
8. If a spoke-SDP in endpoint Y, including the ICB spoke-SDP, received T-LDP SAP Down status bits, or received T-LDP SDP-binding Down status bits, or received status bits of PW not forwarding, the node saves this status and takes no further action. The saved status is used for selecting the active transmit endpoint object as per Pseudowire Redundancy procedures.
9. If all objects in endpoint Y, except the ICB spoke-SDP, transition locally to Down state, or received T-LDP SAP Down status bits, or received T-LDP SDP-binding Down status bits, and/or received status bits of PW not forwarding, the node must send status bits of SDP-binding Down over the X endpoint ICB spoke-SDP only.
10. If all objects in endpoint Y transition locally to Down state, or received T-LDP SAP Down status bits, or received T-LDP SDP-binding Down status bits, or received status bits of PW not forwarding, the node must send status bits of SDP-binding Down over the X endpoint ICB spoke-SDP, and must send a SAP Down notification on the X endpoint SAP via the SAP specific OAM signal in this case the SAP MEP ifStatusTlv operationally-Down and also signal the BGP-MH Site, if this SAP is part of a BGP Site.

Operation for BGP MH Site Support for Ethernet Tunnels

A multi-homed site can be configured on up to four PEs although two PEs are sufficient for most applications with each PE having a single object SAP connecting to the multi-homed site. Note that SR OS G.8031 implementation with load sharing allows multiple PEs as well. The designated forwarder election chooses a single connection to be operationally up with the other placed in standby. Only revertive behavior is supported in this release.

Fate-sharing (the status of one site can be inherited from another site) is achievable using monitor-groups.

The following are supported:

- All Ethernet-tunnels G.8031 SAPs on CE:
  - → 7x50 G.8031 in load sharing mode (recommended)
  - → 7x50 G.8031 in non-load sharing mode
- Epipe and Endpoint with SAPs on PE devices.
- Endpoints with PW.
- Endpoints with active/standby PWs.

There are the following constraints with this feature:

- Not supported with PBB Epipes.
- Spoke SDP (pseudowire).
  - → BGP signaling is not supported.
  - → Cannot use BGP MH for auto-discovered pseudowire. This is achieved in a VPLS service using SHGs, which are not available in Epipes.
- Other multi-chassis redundancy features are not supported on the multi-homed site object, namely:
  - → MC-LAG
  - → MC-EP
  - → MC-ring
  - → MC-APS
- Master and Slave pseudowire is not supported.

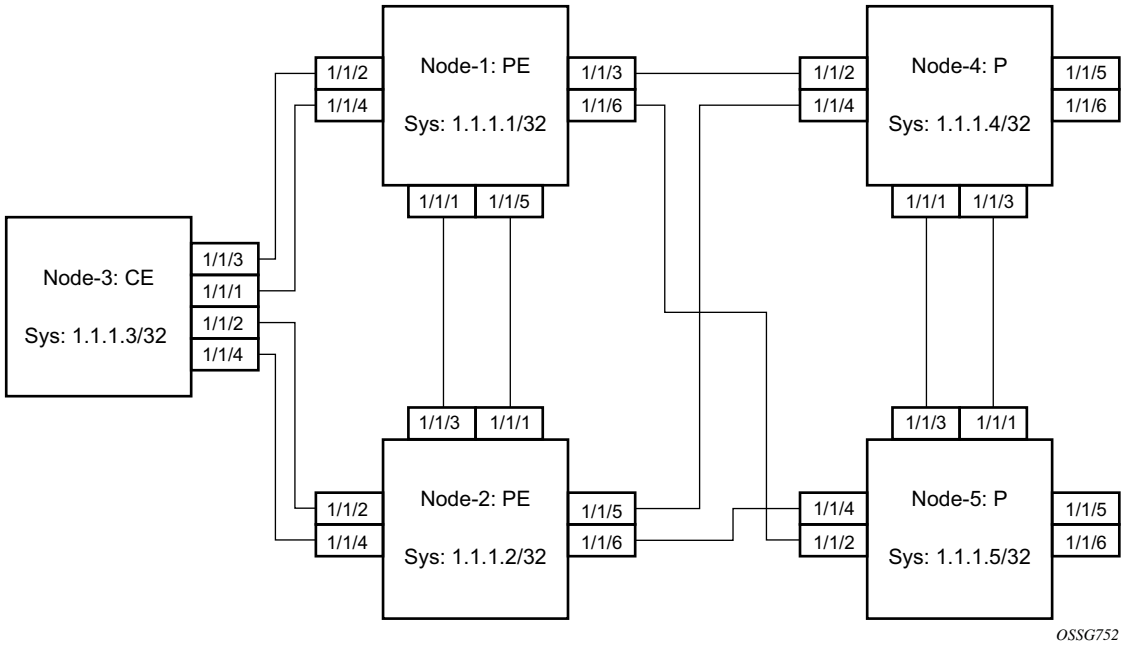


Figure 37: Sample Topology Full Redundancy

Refer to Configuration Examples on page 133 for configuration examples derived from Figure 37.

Configuration Examples

Node-1: Using operational groups and Ethernet CFM per SAP

```
#--------------------------------------------------
echo "Eth-CFM Configuration"
#--------------------------------------------------
    eth-cfm
        domain 100 format none level 3
            association 2 format icc-based name "node-3-site-1-0"
                bridge-identifier 1
                exit
                remote-mepid 310
            exit
            association 2 format icc-based name "node-3-site-1-1"
                bridge-identifier 100
                exit
                remote-mepid 311
            exit
        exit
    exit

#--------------------------------------------------
echo "Service Configuration"
#--------------------------------------------------
    service
        customer 1 create
            description "Default customer"
        exit
        sdp 2 mpls create
            far-end 1.1.1.4
            lsp "to-node-4-lsp-1"
            keep-alive
                shutdown
            exit
            no shutdown
        exit
        sdp 3 mpls create  //  Etcetera

        pw-template 1 create
            vc-type vlan
        exit
        oper-group "og-name-et" create
        exit
        oper-group "og-name-et100" create
        exit
        epipe 1 customer 1 create
            service-mtu 500
            bgp
                route-distinguisher 65000:1
                route-target export target:65000:1 import target:65000:1
            exit
            site "site-1" create
                site-id 1
                sap 1/1/2:1.1
                boot-timer 100
                site-activation-timer 2
                no shutdown
```

```
        exit
        endpoint "x" create
        exit
        endpoint "y" create
        exit
        sap 1/1/2:1.1 endpoint "x" create
            eth-cfm
                mep 130 domain 100 association 2 direction down
                    fault-propagation-enable use-if-tlv
                    ccm-enable
                    no shutdown
                exit
            exit
            oper-group "og-name-et"
        exit
        spoke-sdp 2:1 endpoint "y" create
            precedence primary
            no shutdown
        exit
        spoke-sdp 3:1 endpoint "y" create
            precedence 2
            no shutdown
        exit
        no shutdown
    exit
    epipe 100 customer 1 create
        description "Epipe 100 in separate opergroup"
        service-mtu 500
        bgp
            route-distinguisher 65000:2
            route-target export target:65000:2 import target:65000:2
        exit
        site "site-name-et100" create
            site-id 1101
            sap 1/1/4:1.100
            boot-timer 100
            site-activation-timer 2
            no shutdown
        exit

        endpoint "x" create
        exit
        endpoint "y" create
        exit
        sap 1/1/4:1.100 endpoint "x" create
                eth-cfm
                mep 131 domain 1 association 2 direction down
                    fault-propagation-enable use-if-tlv
                    ccm-enable
                    no shutdown
                exit
            exit
            oper-group "og-name-et100"

        exit
        spoke-sdp 2:2 vc-type vlan endpoint "y" create
            precedence 1
            no shutdown
        exit
```

```
            spoke-sdp 3:2 vc-type vlan endpoint "y" create
                precedence 2
                no shutdown
            exit
            no shutdown
        exit

    exit
#--------------------------------------------------
echo "BGP Configuration"
#--------------------------------------------------
        bgp
            rapid-withdrawal
            rapid-update l2-vpn
            group "internal"
                type internal
                neighbor 1.1.1.2
                    family l2-vpn
                exit
            exit
        exit
    exit
```

Node-3: Using operational groups and Ethernet CFM per SAP

```
#--------------------------------------------------
echo "Eth-CFM Configuration"
#--------------------------------------------------
    eth-cfm
        domain 100 format none level 3
            association 2 format icc-based name "node-3-site-1-0"
                bridge-identifier 1
                exit
                ccm-interval 1
                remote-mepid 130
            exit
            association 2 format icc-based name "node-3-site-1-1"
                bridge-identifier 100
                exit
                ccm-interval 1
                remote-mepid 131
            association 3 format icc-based name "node-3-site-2-0"
                bridge-identifier 1
                exit
                ccm-interval 1
                remote-mepid 120
            exit
            association 3 format icc-based name "node-3-site-2-1"
                bridge-identifier 100
                exit
                ccm-interval 1
                remote-mepid 121
            exit
        exit
    exit
```

```
#--------------------------------------------------
echo "Service Configuration"
#--------------------------------------------------

  eth-tunnel 1
        description "Eth Tunnel loadsharing mode QinQ example"
        protection-type loadsharing
        ethernet
            encap-type qinq
        exit
        path 1
            member 1/1/3
            control-tag 1.1
            eth-cfm
                mep 310 domain 100 association 2
                    ccm-enable
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        path 2
            member 1/1/4
            control-tag 1.2
            eth-cfm
                mep 320 domain 100 association 3
                    ccm-enablepath
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        no shutdown
    exit
#--------------------------------------------------
echo "Ethernet Tunnel Configuration"
#--------------------------------------------------
    eth-tunnel 2
        description "Eth Tunnel QinQ"
        revert-time 10
        path 1
            precedence primary
            member 1/1/1
            control-tag 1.100
            eth-cfm
                mep 311 domain 100 association 2
                    ccm-enable
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        path 2
            member 1/1/2
            control-tag 1.100
            eth-cfm
```

```
                mep 321 domain 100 association 3
                    ccm-enable
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        no shutdown
    exit
#--------------------------------------------------
echo "Service Configuration"
#--------------------------------------------------
    service
        epipe 1 customer 1 create
            sap 2/1/2:1.1 create
            exit
            sap eth-tunnel-1 create
            exit
            no shutdown
        exit
        epipe 100 customer 1 create
            service-mtu 500
            sap 2/1/10:1.100 create
            exit
            sap eth-tunnel-2 create
            exit
            no shutdown
        exit
```

Configuration with Fate Sharing on Node-3 In this example the SAPs monitoring the operational groups do not need CFM if the corresponding SAP on the CE side is using fate sharing.

Node-1:

```
#--------------------------------------------------
echo "Service Configuration"  Oper-groups
#--------------------------------------------------
    service
        customer 1 create
            description "Default customer"
        exit
        sdp 2 mpls create
          ...

        exit
        pw-template 1 create
            vc-type vlan
        exit
        oper-group "og-name-et" create
        exit
        epipe 1 customer 1 create
            service-mtu 500
            bgp
                route-distinguisher 65000:1
                route-target export target:65000:1 import target:65000:1
```

```
            exit
            site "site-1" create
                site-id 1
                sap 1/1/2:1.1
                boot-timer 100
                site-activation-timer 2
                no shutdown
            exit
            endpoint "x" create
            exit
            endpoint "y" create
            exit
            sap 1/1/2:1.1 endpoint "x" create
                eth-cfm
                    mep 130 domain 100 association 1 direction down
                        fault-propagation-enable use-if-tlv
                        ccm-enable
                        no shutdown
                    exit
                exit
                oper-group "og-name-et"
            exit
            spoke-sdp 2:1 endpoint "y" create
                precedence primary
                no shutdown
            exit
            spoke-sdp 3:1 endpoint "y" create
                precedence 2
                no shutdown
            exit
            no shutdown
        exit
        epipe 2 customer 1 create
            description "Epipe 2 in opergroup with Epipe 1"
            service-mtu 500
            bgp
                route-distinguisher 65000:2
                route-target export target:65000:2 import target:65000:2
            exit
            endpoint "x" create
            exit
            endpoint "y" create
            exit
            sap 1/1/2:1.2 endpoint "x" create
                monitor-oper-group "og-name-et"
            exit
            spoke-sdp 2:2 vc-type vlan endpoint "y" create
                precedence 1
                no shutdown
            exit
            spoke-sdp 3:2 vc-type vlan endpoint "y" create
                precedence 2
                no shutdown
            exit
            no shutdown
        exit

    exit
```

Node-3:

```
#--------------------------------------------------
echo "Eth-CFM Configuration"
#--------------------------------------------------
    eth-cfm
        domain 100 format none level 3
            association 1 format icc-based name "node-3-site-1-0"
                bridge-identifier 1
                exit
                ccm-interval 1
                remote-mepid 130
            exit
            association 2 format icc-based name "node-3-site-2-0"
                bridge-identifier 2
                exit
                ccm-interval 1
                remote-mepid 120
            exit
        exit
    exit

#--------------------------------------------------
echo "Service Configuration"
#--------------------------------------------------

  eth-tunnel 2
        description "Eth Tunnel loadsharing mode QinQ example"
        protection-type loadsharing
        ethernet
            encap-type qinq
        exit
        path 1
            member 1/1/1
            control-tag 1.1
            eth-cfm
                mep 310 domain 100 association 1
                    ccm-enable
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        path 2
            member 1/1/2
            control-tag 1.1
            eth-cfm
                mep 320 domain 100 association 2
                    ccm-enablepath
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        no shutdown
    exit
```

```
#--------------------------------------------------
echo "Service Configuration"
#--------------------------------------------------
    service
        epipe 1 customer 1 create
            sap 1/10/1:1 create
            exit
            sap eth-tunnel-1 create
            exit
            no shutdown
        exit
#--------------------------------------------------
echo "Service Configuration for a shared fate Ethernet Tunnel"
#--------------------------------------------------
        epipe 2 customer 1 create
            sap 1/10/2:3 create
            exit
            sap eth-tunnel-1:2 create
                eth-tunnel
                    path 1 tag 1.2
                    path 2 tag 1.2
                exit
            exit
            no shutdown
        exit
```

Access Node Resilience Using MC-LAG and Pseudowire Redundancy

Figure 38 shows the use of both Multi-Chassis Link Aggregation (MC-LAG) in the access network and pseudowire redundancy in the core network to provide a resilient end-to-end VLL service to the customers.

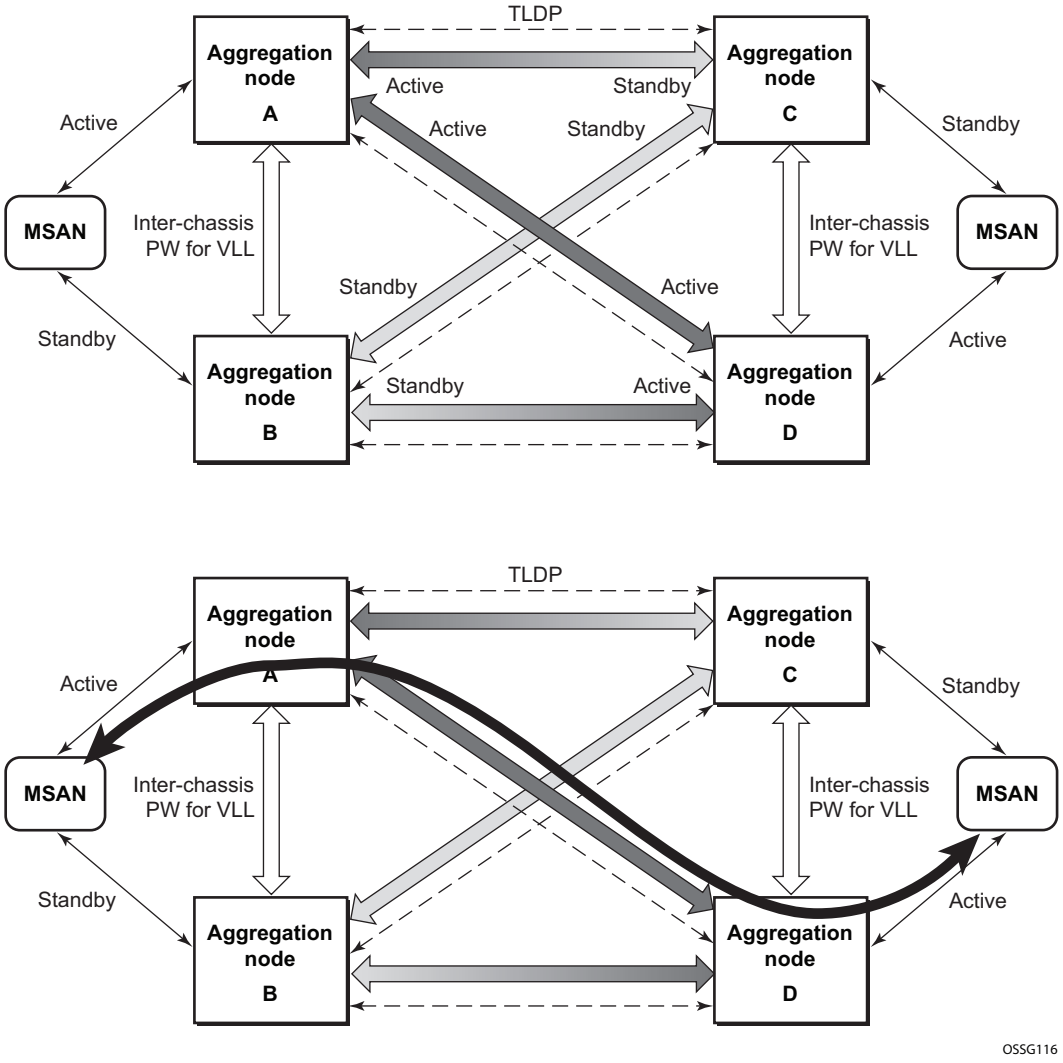


Figure 38: Access Node Resilience

In this application, a new pseudowire status bit of active or standby indicates the status of the SAP in the MC-LAG instance in the SR-Series aggregation node. All spoke SDPs are of secondary type and there is no use of a primary pseudowire type in this mode of operation. Node A is in the active

state according to its local MC-LAG instance and thus advertises active status notification messages to both its peer pseudowire nodes, for example, nodes C and D. Node D performs the same operation. Node B is in the standby state according to the status of the SAP in its local MC-LAG instance and thus advertises standby status notification messages to both nodes C and D. Node C performs the same operation.

An SR-Series node selects a pseudowire as the active path for forwarding packets when both the local pseudowire status and the received remote pseudowire status indicate active status. However, an SR-Series device in standby status according to the SAP in its local MC-LAG instance is capable of processing packets for a VLL service received over any of the pseudowires which are up. This is to avoid black holing of user traffic during transitions. The SR-Series standby node forwards these packets to the active node bye the Inter-Chassis Backup pseudowire (ICB pseudowire) for this VLL service. An ICB is a spoke SDP used by a MC-LAG node to backup a MC-LAG SAP during transitions. The same ICB can also be used by the peer MC-LAG node to protect against network failures causing the active pseudowire to go down.

Note that at configuration time, the user specifies a precedence parameter for each of the pseudowires which are part of the redundancy set as described in the application in VLL Resilience with Two Destination PE Nodes on page 112. An SR-Series node uses this to select which pseudowire to forward packet to in case both pseudowires show active/active for the local/remote status during transitions.

Only VLL service of type Epipe is supported in this application. Furthermore, ICB spoke SDP can only be added to the SAP side of the VLL cross-connect if the SAP is configured on a MC-LAG instance.

VLL Resilience for a Switched Pseudowire Path

Figure 39 illustrates the use of both pseudowire redundancy and pseudowire switching to provide a resilient VLL service across multiple IGP areas in a provider network.

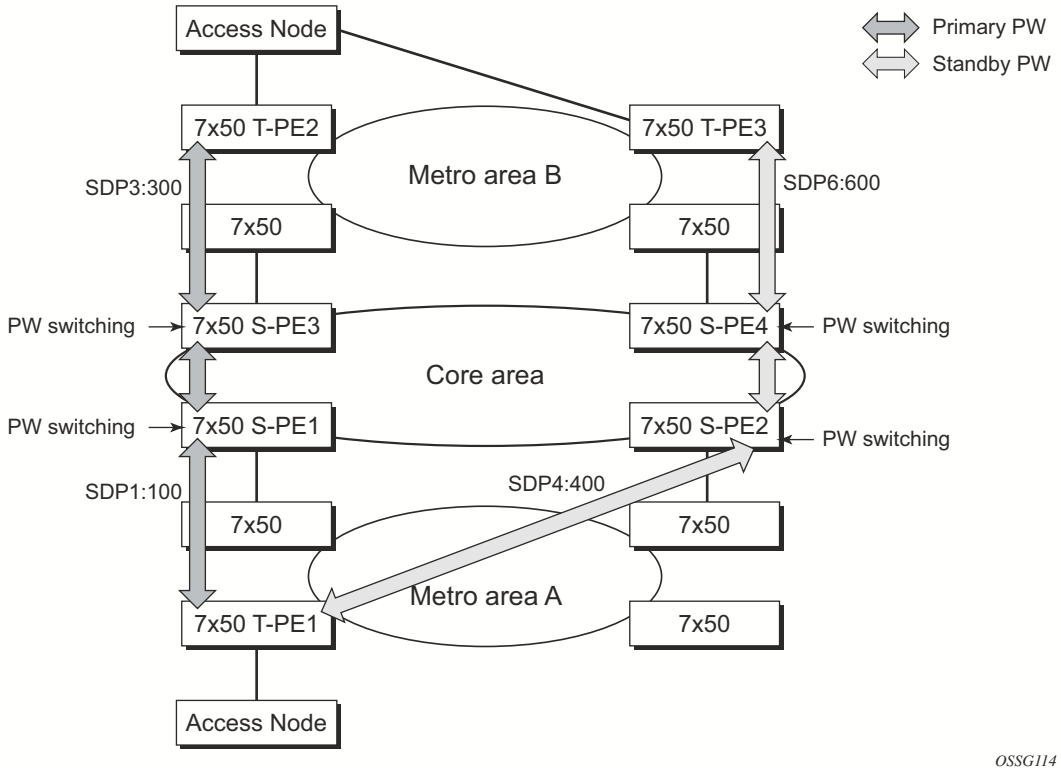


Figure 39: VLL Resilience with Pseudowire Redundancy and Switching

Pseudowire switching is a method for scaling a large network of VLL or VPLS services by removing the need for a full mesh of T-LDP sessions between the PE nodes as the number of these nodes grows over time.

Like in the application in VLL Resilience with Two Destination PE Nodes on page 112, the T-PE1 node switches the path of a VLL to a secondary standby pseudowire in the case of a network side failure causing the VLL binding status to be DOWN or if T-PE2 notified it that the remote SAP went down. This application requires that pseudowire status notification messages generated by either a T-PE node or a S-PE node be processed and relayed by the S-PE nodes.

Note that it is possible that the secondary pseudowire path terminates on the same target PE as the primary, for example, T-PE2. This provides protection against network side failures but not against a remote SAP failure. When the target destination PE for the primary and secondary

pseudowires is the same, T-PE1 will normally not switch the VLL path onto the secondary pseudowire upon receipt of a pseudowire status notification indicating the remote SAP is down since the status notification is sent over both the primary and secondary pseudowires. However, the status notification on the primary pseudowire may arrive earlier than the one on the secondary pseudowire due to the differential delay between the paths. This will cause T-PE1 to switch the path of the VLL to the secondary standby pseudowire and remain there until the status notification is cleared. At that point in time, the VLL path is switched back to the primary pseudowire due to the revertive behavior operation. The path will not switch back to a secondary path when it becomes up even if it has a higher precedence than the currently active secondary path.

This application can make use of all types of VLL supported on the routers, for example, Apipe, Fpipe, Epipe, and Ipipe services. A SAP can be configured on SONET/SDH port which is part of an APS group. However, if a SAP is configured on a MC-LAG instance, only the Epipe service type will be allowed.

Pseudowire Redundancy Service Models

This section describes the various MC-LAG and pseudowire redundancy scenarios as well as the algorithm used to select the active transmit object in a VLL endpoint.

The redundant VLL service model is described in the following section, Redundant VLL Service Model.

Redundant VLL Service Model

In order to implement pseudowire redundancy, a VLL service accommodates more than a single object on the SAP side and on the spoke SDP side. Figure 40 illustrates the model for a redundant VLL service based on the concept of endpoints.

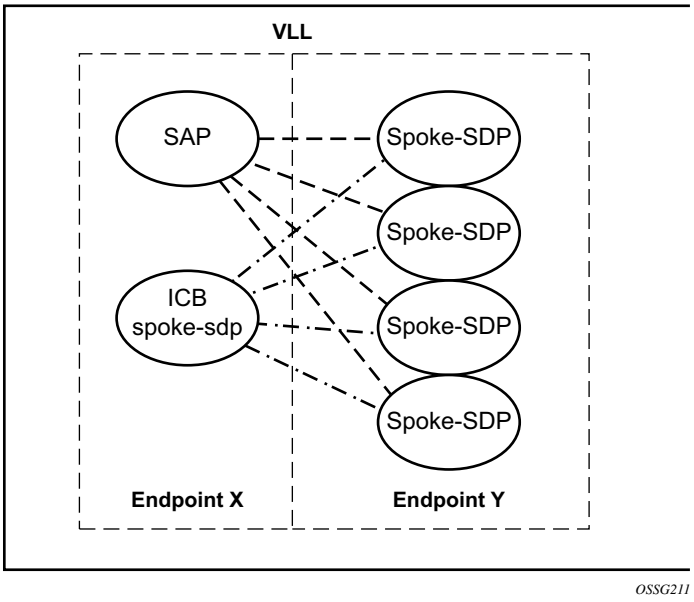


Figure 40: Redundant VLL Endpoint Objects

A VLL service supports by default two implicit endpoints managed internally by the system. Each endpoint can only have one object, a SAP or a spoke SDP.

In order to add more objects, up to two (2) explicitly named endpoints may be created per VLL service. The endpoint name is locally significant to the VLL service. They are referred to as endpoint 'X' and endpoint 'Y' as illustrated in Figure 40.

Note that Figure 40 is merely an example and that the "Y" endpoint can also have a SAP and/or an ICB spoke SDP. The following details the four types of endpoint objects supported and the rules used when associating them with an endpoint of a VLL service:

- SAP — There can only be a maximum of one SAP per VLL endpoint.
- Primary spoke SDP — The VLL service always uses this pseudowire and only switches to a secondary pseudowire when it is down the VLL service switches the path to the primary pseudowire when it is back up. The user can configure a timer to delay reverting back to primary or to never revert. There can only be a maximum of one primary spoke SDP per VLL endpoint.
- Secondary spoke SDP — There can be a maximum of four secondary spoke SDP per endpoint. The user can configure the precedence of a secondary pseudowire to indicate the order in which a secondary pseudowire is activated.
- Inter-Chassis Backup (ICB) spoke SDP — Special pseudowire used for MC-LAG and pseudowire redundancy application. Forwarding between ICBs is blocked on the same node. The user has to explicitly indicate the spoke SDP is actually an ICB at creation time. There are however a few scenarios below where the user can configure the spoke SDP as ICB or as a regular spoke SDP on a given node. The CLI for those cases will indicate both options.

A VLL service endpoint can only use a single active object to transmit at any given time but can receive from all endpoint objects

An explicitly named endpoint can have a maximum of one SAP and one ICB. Once a SAP is added to the endpoint, only one more object of type ICB spoke SDP is allowed. The ICB spoke SDP cannot be added to the endpoint if the SAP is not part of a MC-LAG instance. Conversely, a SAP which is not part of a MC-LAG instance cannot be added to an endpoint which already has an ICB spoke SDP.

An explicitly named endpoint, which does not have a SAP object, can have a maximum of four spoke SDPs and can include any of the following:

- A single primary spoke SDP.
- One or many secondary spoke SDPs with precedence.
- A single ICB spoke SDP.

T-LDP Status Notification Handling Rules

Referring to Figure 40 on page 145 as a reference, the following are the rules for generating, processing, and merging T-LDP status notifications in VLL service with endpoints. Note that any allowed combination of objects as specified in Redundant VLL Service Model on page 145 can be used on endpoints “X” and “Y”. The following sections refer to the specific combination objects in Figure 40 as an example to describe the more general rules.

Processing Endpoint SAP Active/Standby Status Bits

The advertised admin forwarding status of active/standby reflects the status of the local LAG SAP in MC-LAG application. If the SAP is not part of a MC-LAG instance, the forwarding status of active is always advertised.

When the SAP in endpoint “X” is part of a MC-LAG instance, a node must send T-LDP forwarding status bit of “SAP active/standby” over all “Y” endpoint spoke SDPs, except the ICB spoke SDP, whenever this status changes. The status bit sent over the ICB is always zero (active by default).

When the SAP in endpoint “X” is not part of a MC-LAG instance, then the forwarding status sent over all “Y” endpoint spoke SDP's should always be set to zero (active by default).

Processing and Merging

Endpoint “X” is operationally up if at least one of its objects is operationally up. It is down if all its objects are operationally down.

If the SAP in endpoint “X” transitions locally to the down state, or received a SAP down notification by SAP-specific OAM signal, the node must send T-LDP SAP down status bits on the “Y” endpoint ICB spoke SDP only. Note that Ethernet SAP does not support SAP OAM protocol. All other SAP types cannot exist on the same endpoint as an ICB spoke SDP since non Ethernet SAP cannot be part of a MC-LAG instance.

If the ICB spoke SDP in endpoint “X” transitions locally to down state, the node must send T-LDP SDP-binding down status bits on this spoke SDP.

If the ICB spoke SDP in endpoint “X” received T-LDP SDP-binding down status bits or pseudowire not forwarding status bits, the node saves this status and takes no further action. The saved status is used for selecting the active transmit endpoint object.

If all objects in endpoint “X” transition locally to down state, and/or received a SAP down notification by remote T-LDP status bits or by SAP specific OAM signal, and/or received status

bits of SDP-binding down, and/or received status bits of pseudowire not forwarding, the node must send status bits of SAP down over all “Y” endpoint spoke SDPs, including the ICB.

Endpoint “Y” is operationally up if at least one of its objects is operationally up. It is down if all its objects are operationally down.

If a spoke SDP in endpoint “Y”, including the ICB spoke SDP, transitions locally to down state, the node must send T-LDP SDP-binding down status bits on this spoke SDP.

If a spoke SDP in endpoint “Y”, including the ICB spoke SDP, received T-LDP SAP down status bits, and/or received T-LDP SDP-binding down status bits, and/or received status bits of pseudowire not forwarding, the node saves this status and takes no further action. The saved status is used for selecting the active transmit endpoint object.

If all objects in endpoint “Y”, except the ICB spoke SDP, transition locally to down state, and/or received T-LDP SAP down status bits, and/or received T-LDP SDP-binding down status bits, and/or received status bits of pseudowire not forwarding, the node must send status bits of SDP-binding down over the “X” endpoint ICB spoke SDP only.

If all objects in endpoint “Y” transition locally to down state, and/or received T-LDP SAP down status bits, and/or received T-LDP SDP-binding down status bits, and/or received status bits of pseudowire not forwarding, the node must send status bits of SDP-binding down over the “X” endpoint ICB spoke SDP, and must send a SAP down notification on the “X” endpoint SAP by the SAP specific OAM signal if applicable. An Ethernet SAP does not support signaling status notifications.

High-Speed Downlink Packet Access (HSDPA) Off Load Fallback over ATM

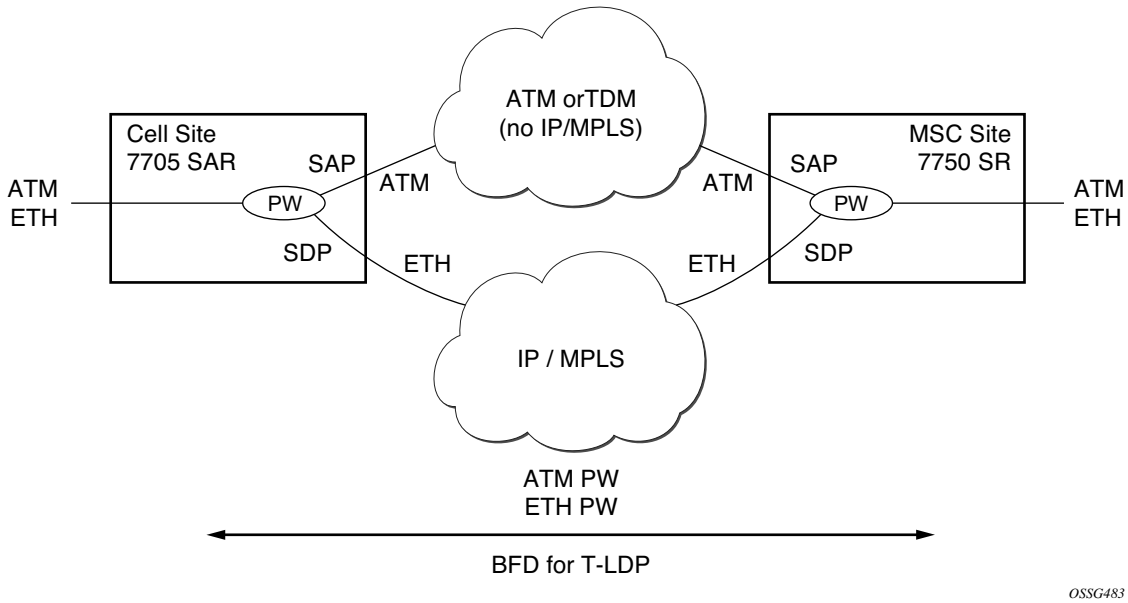


Figure 41: HSDPA Off Load Fallback over ATM

For many Universal Mobile Telecommunications System (UMTS) networks planning to deploy High-Speed Downlink Packet Access (HSDPA), the existing mobile backhaul topology consists of a cell site that is partially backhauled over DSL (for the HSDPA portion) and partially over an existing TDM/ATM infrastructure (for UMTS voice traffic).

For example, the service pseudowires provider may use a 7705 SAR with one or two ATM E1 uplinks for real-time voice traffic and an Ethernet uplink connected to a DSL model for NRT data traffic. At the RNC site, a 7750 SR service router can be used, connected by ASAP (E1 IMA bundles) or STM-n ATM to the TDM/ATM network, and Ethernet to the DSL backhaul network.

On the MSC-located SR connected to the Radio Network Controller (RNC), there is a standard pseudowire (Ethernet or ATM) which has an active pseudowire by IP/MPLS, but the standby path is not IP/MPLS capable Therefore, the active/standby pseudowire concept is extended to allow standby to be an access SAP to an ATM network for ATM pseudowire or Ethernet (bridged over ATM) for ETH pseudowire.

Normally, if the MPLS pseudowire path is active, this is taken. If a failure happens on the IP/MPLS path, detected through BFD-TLDP or local notification, we need to switch to the SAP which is connected to the ATM/TDM backhaul network As soon as the MPLS pseudowire path becomes available again, reversion back to the pseudowire path is supported.

Primary Spoke-SDP Fallback to Secondary SAP

For HSDPA, Apipe and Epipe service termination on the SR where an endpoint-X SAP connects to the mobile RNC (by ATM or Ethernet) and an endpoint Y has a primary spoke SDP and a secondary SAP on an SR ATM or ASAP MDA (with bridged PDU encapsulation for Epipes). The secondary SAP has the same restrictions as the SAP in endpoint-X for Apipe and Epipe respectively.

It sufficient to have a single secondary SAP (without any precedence) which implies it can not be mixed with any secondary spoke-SDPs 1+1 APS and MC-APS is supported on the secondary SAP interface.

Similar to the current pseudowire redundancy implementation, receive should be enabled on both objects even though transmit is only enabled on one.

It is expected that BFD for T-LDP [bfd-for-tldp] will be used in most applications to decrease the fault detection times and minimize the outage times upon failure.

Reversion to Primary Spoke SDP Path

The **endpoint revert-time** reversion from secondary to primary paths in the **config>service>apipe>endpoint** and **config>service>epipe>endpoint** contexts are consistent with standard pseudowire redundancy. Various network configurations and equipment require different reversion configurations. The default revert-time is 0.

MC-APS and MC-LAG

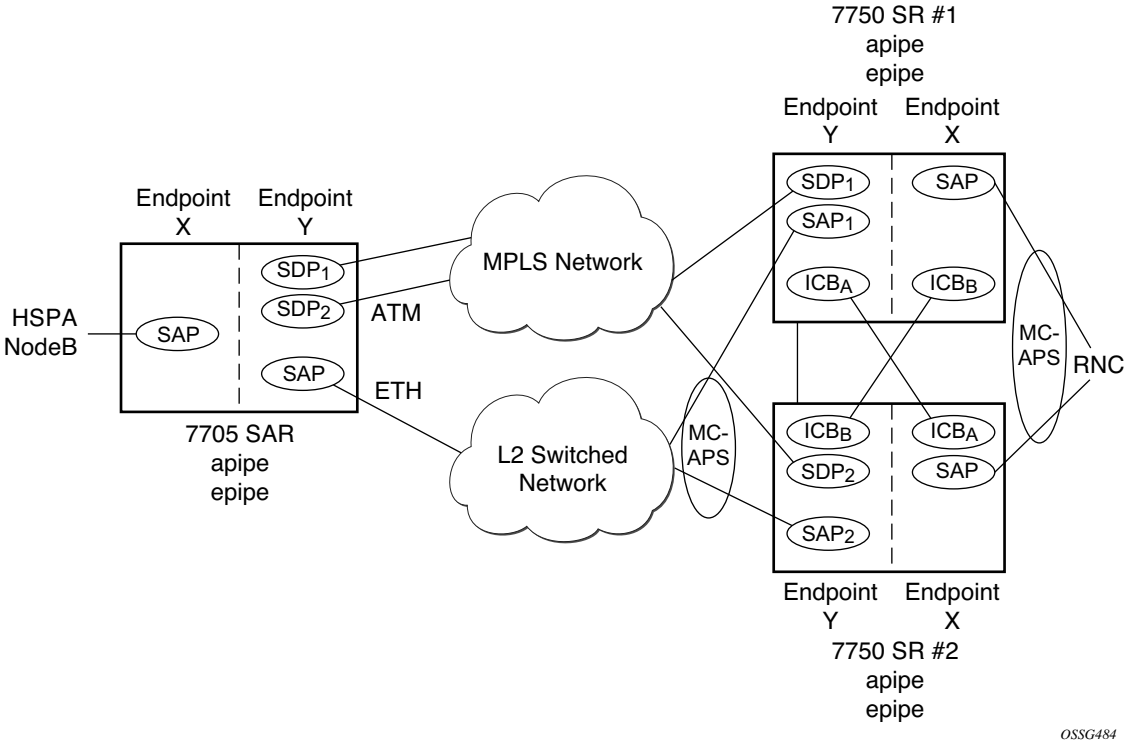


Figure 42: HSDPA Off Load Fallback with MC-APS

In many cases, 7750 SRs are deployed in redundant pairs at the MSC. In this case, MC-APS is typically used for all ATM connections. Figure 42 illustrates this case assuming that MC-APS is deployed on both the RNC connection and the ATM network connection. For MC-APS to be used, clear channel SONET or SDH connections should be used.

In this scenario, endpoint Y allows the addition of an ICB spoke SDP in addition to the primary spoke SDP and secondary SAP. ICB operation is maintained as the current redundant pseudowire operation and the ICB spoke SDP is always given an active status. The ICB spoke SDP is only used if both the primary spoke SDP and secondary SAP are not available. The secondary SAP is used if it is operationally up and the primary spoke SDP pseudowire status is not active. The receive is enabled on all objects even though transmit is only enabled on one.

To allow proper operation in all failure scenarios, an ICB spoke SDP must be added to endpoint X. The ICB spoke SDP is only used if the SAP is operationally down.

The following is an example configuration of Epipes mapping to Figure 42. Note that a SAP can be added to an endpoint with a non-ICB spoke SDP only if the spoke's precedence is **primary**.

7750 SR#1:

```
*A:ALA-A>config>service#  epipe 1
-----------------------------------------------
            endpoint X
            exit
            endpoint Y
            exit
            sap 1/1/2:0 endpoint X
              exit
            spoke-sdp 1:100 endpoint X icb
              exit
            spoke-sdp 10:500 endpoint Y
              precedence primary
              exit
            sap 1/1/3:0 endpoint Y
            exit
            spoke-sdp 1:200 endpoint Y icb
            exit
-----------------------------------------------
*A:ALA-A>config>service#
```

7750 SR#2

```
*A:ALA-B>config>service# epipe 1
-----------------------------------------------
            endpoint X
            exit
            endpoint Y
            exit
            sap 2/3/4:0 endpoint X
            exit
            spoke-sdp 1:200 endpoint X icb
            exit
            spoke-sdp 20:600 endpoint Y
              precedence primary
              exit
            sap 2/3/5:0 endpoint Y
            exit
            spoke-sdp 1:100 endpoint Y icb
            exit
-----------------------------------------------
*A:ALA-B>config>service#
```

Failure Scenarios

Following the before mentioned rules, the following are examples of a failure scenario operation. Assuming both links are active on 7750 SR#1 and the Ethernet connection to the cell site fails (most likely failure scenario as it would not be protected), SDP1 would go down and the secondary SAP would be used in 7750 SR#1 and 7705 SAR as shown in Figure 43.

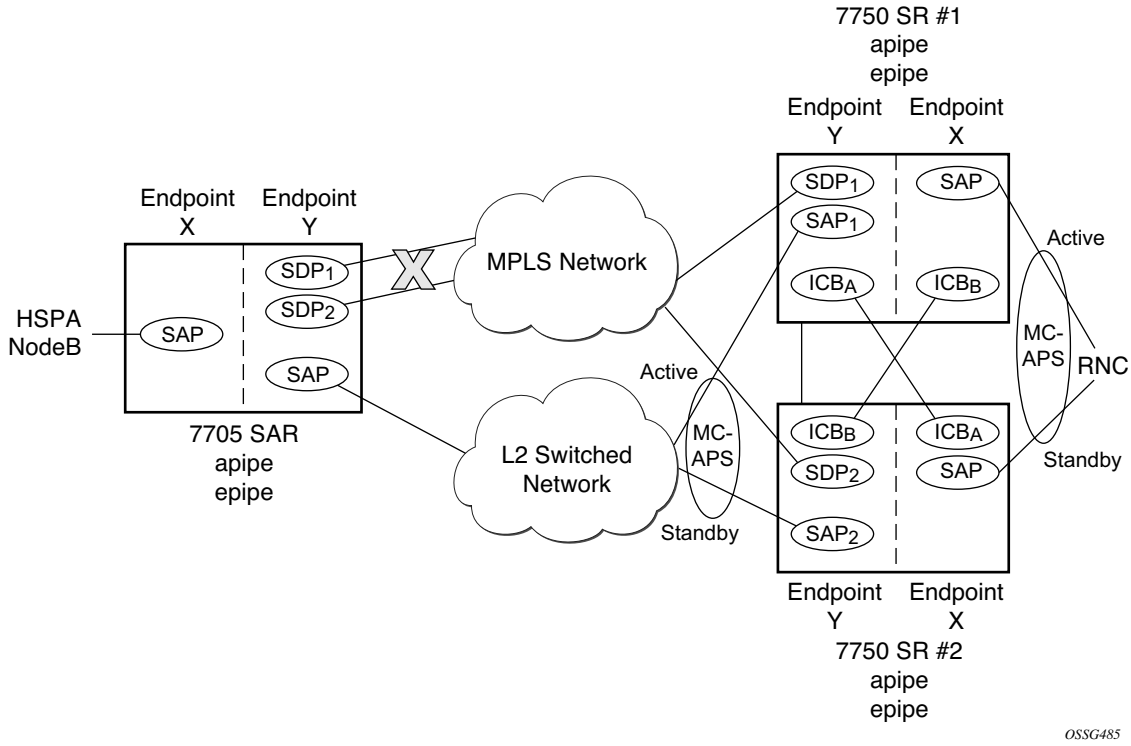


Figure 43: Ethernet Failure At Cell Site

If the active link to the Layer 2 switched network was on 7750 SR#2 at the time of the failure, SAP1 would be operationally down (as the link is in standby) and ICBA would be used. As the RNC SAP on 7750 SR#2 is on a standby APS link, ICBA would be active and it would connect to SAP2 as SDP2 is operationally down as well.

All APS link failures would be handled through the standard pseudowire status messaging procedures for the RNC connection and through standard ICB usage for the Layer 2 switched network connection.

VLL Using G.8031 Protected Ethernet Tunnels

The use of MPLS tunnels provides a way to scale the core while offering fast failover times using MPLS FRR. In environments where Ethernet services are deployed using native Ethernet backbones, Ethernet tunnels are provided to achieve the same fast failover times as in the MPLS FRR case.

The Alcatel-Lucent VLL implementation offers the capability to use core Ethernet tunnels compliant with ITU-T G.8031 specification to achieve 50 ms resiliency for backbone failures. This is required to comply with the stringent SLAs provided by service providers in the current competitive environment. Epipe and Ipipe services are supported.

When using Ethernet Tunnels, the Ethernet Tunnel logical interface is created first. The Ethernet tunnel has member ports which are the physical ports supporting the links. The Ethernet tunnel control SAPs carries G.8031 and 802.1ag control traffic and user data traffic. Ethernet service SAPs are configured on the Ethernet tunnel. Optionally when tunnels follow the same paths end to end services may be configured with, same-fate Ethernet tunnel SAPs which carry only user data traffic and shares the fate of the Ethernet tunnel port (if properly configured).

Ethernet tunnels provide a logical interface that VLL SAPs may use just as regular interfaces. The Ethernet tunnel provides resiliency by providing end to end tunnels. The tunnels are stitched together by VPLS or Epipe services at intermediate points. Epipes offer a more scalable option.

For further information, see the *Services Overview Guide*.

BGP Virtual Private Wire Service (VPWS)

BGP Virtual Private Wire Service (VPWS) is a point-to-point L2 VPN service based on RFC 6624 (Layer 2 Virtual Private Networks using BGP for Auto-Discovery and Signaling) which in turn uses the BGP pseudowire signaling concepts from RFC 4761, *Virtual Private LAN Service Using BGP for Auto-Discovery and Signaling*.

Single-Homed BGP VPWS

A single-homed BGP VPWS service is implemented as an Epipe connecting a SAP or static GRE tunnel (a spoke SDP using a GRE SDP configured with static MPLS labels) and a BGP signaled pseudowire, maintaining the Epipe properties such as no MAC learning. The pseudowire data plane uses a two label stack, the inner label is derived from the BGP signaling and identifies the Epipe service while the outer label is the tunnel label of an LSP transporting the traffic between the two end systems.

Figure 44 shows how this service would be used to provide a virtual lease-line service across an MPLS network between two sites, A and B.

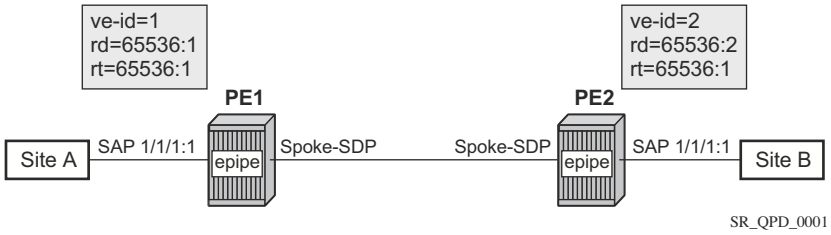


Figure 44: Single-Homed BGP-VPWS Example

An Epipe is configured on PE1 and PE2 with BGP VPWS enabled. PE1 and PE2 are connected to site A and B, respectively, each using a SAP. The interconnection between the two PEs is achieved through a pseudowire which is signaled using BGP VPWS updates over a given tunnel LSP.

Dual-Homed BGP VPWS

A BGP-VPWS service can benefit from dual-homing, as described in draft-ietf-l2vpn-vpls-multihoming-03. When using dual-homing, two PEs connect to a site with one PE being the designated forwarder for the site and the other blocking its connection to the site. On failure of the active PE, its pseudowire or its connection to the site, the other PE becomes the designated forwarder and unblocks its connection to the site.

Single Pseudowire Example:

A pseudowire is established between the designated forwarder of the dual-homed PEs and the remote PE. If a failure causes a change in the designated forwarder, the pseudowire is deleted and re-established between the remote PE and the new designated forwarder. This topology requires that the VE IDs on the dual-homed PEs are set to the same value.

An example is shown in Figure 45.

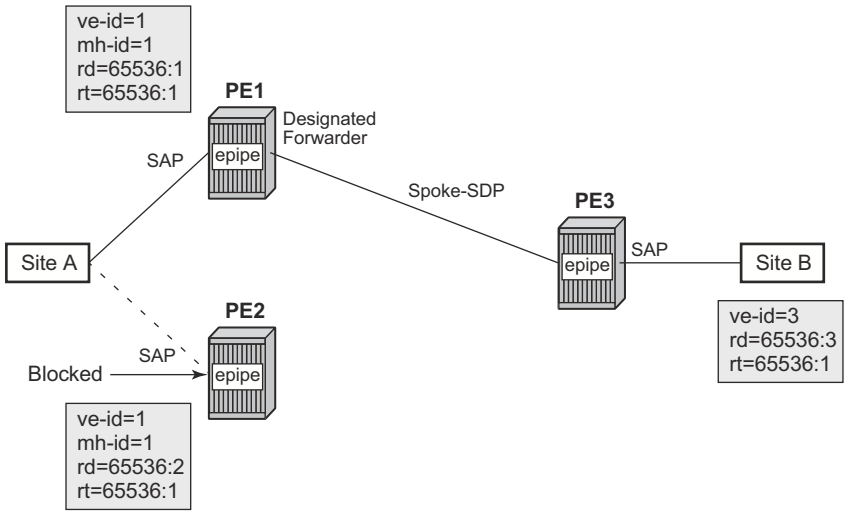


Figure 45: Dual-Homed BGP VPWS with Single Pseudowire

An Epipe with BGP VPWS enabled is configured on each PE. Site A is dual-homed to PE1 and PE2 with the remote PE, PE3, connecting to site B. An Epipe service is configured on each PE in which there is a SAP connecting to the local site.

The pair of dual-homed PEs perform a designated forwarder election, which is influenced by BGP route selection, the site state, and by configuring the site-preference. A site will only be eligible to be the designated forwarder if it is up (note that the site state will be down if there is no pseudowire established or if the pseudowire is in an oper down state). The winner, for example PE1, becomes the active switch for traffic sent to and from site A, while the loser blocks its

connection to site A. Pseudowires are signaled using BGP from PE1 and PE2 to PE3 but only from PE3 to the designated forwarder in the opposite direction (thereby only one bi-directional pseudowire is established). There is no pseudowire between PE1 and PE2; this is achieved by configuration.

Traffic is sent and received traffic on the pseudowire connected between PE3 and the designated forwarder, PE1.

If the site state is oper down then both the D and CSV bits (see below for more details) are set in the BGP-VPWS update which will cause the remote PE to use the pseudowire to the new designated forwarder.

Active/Standby Pseudowire Example:

Pseudowires are established between the remote PE and each dual-homed PE. The remote PE can receive traffic on either pseudowire but will only send on the one to the designated forwarder. This creates an active/standby pair of pseudowires. At most one standby pseudowire will be established; this being determined using the tie-breaking rules defined in the multi-homing draft. This topology requires each PE to have a different VE ID.

A dual-homed topology example is shown in Figure 46.

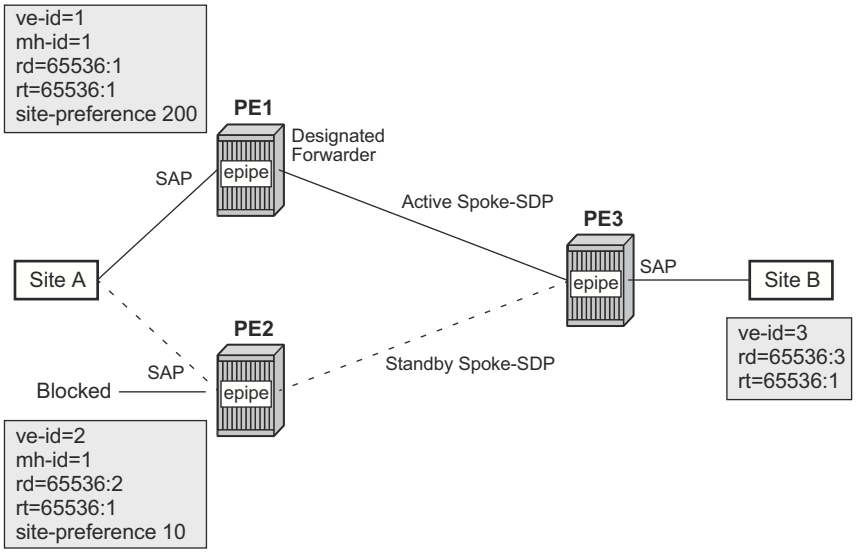


Figure 46: Dual-homed BGP VPWS with Active/Standby Pseudowires

An Epipe with BGP VPWS enabled is configured on each PE. Site A is dual-homed to PE1 and PE2 with the remote PE, PE3, connecting to site B. An Epipe service is configured on each PE in which there is a SAP connecting to the local site.

The pair of dual-homed PEs perform a designated forwarder election, which is influenced by configuring the site-preference. The winner, PE1 (based on its higher site-preference) becomes the active switch for traffic sent to and from site A, while the loser, PE2, blocks its connection to site A. Pseudowires are signaled using BGP between PE1 and PE3, and between PE2 and PE3. There is no pseudowire between PE1 and PE2; this is achieved by configuration. The active/standby pseudowires on PE3 are part of an endpoint automatically created in the Epipe service.

Traffic is sent and received traffic on the pseudowire connected to the designated forwarder, PE1.

BGP VPWS Pseudowire Switching

Pseudowire switching is supported with a BGP VPWS service allowing the cross connection between a BGP VPWS signaled spoke SDP and a static GRE tunnel, the latter being a spoke SDP configured with static MPLS labels using a GRE SDP. No other spoke SDP types are supported. Support is not included for BGP multi-homing using an active and a standby pseuodowire to a pair of remote PEs.

Operational state changes to the GRE tunnel are reflected in the state of the Epipe and propagated accordingly in the BGP VPWS spoke SDP's status signaling, specifically using the BGP update D/csv bits.

The following configuration is required:

1. The Epipe service must be created using the **vc-switching** parameter.
2. The GRE tunnel spoke SDP must be configured using a GRE SDP with **signaling off**, and have the ingress and egress vc-labels statically configured.

An example configuration is shown below:

```
configure
    service
        sdp 1 create
            signaling off
            far-end 192.168.1.1
            keep-alive
                shutdown
            exit
            no shutdown
        exit
        pw-template 1 create
        exit
        epipe 1 customer 1 vc-switching create
            description "BGP VPWS service"
            bgp
                route-distinguisher 65536:1
                route-target export target:65536:1 import target:65536:1
                pw-template-binding 1
                exit
            exit
```

```
        bgp-vpws
            ve-name "PE1"
                ve-id 1
            exit
            remote-ve-name "PE2"
                ve-id 2
            exit
            no shutdown
        exit
        spoke-sdp 1:1 create
            ingress
                vc-label 1111
            exit
            egress
                vc-label 1122
            exit
            no shutdown
        exit
        no shutdown
    exit
```

Pseudowire Signaling

The BGP signaling mechanism used to establish the pseudowires is described in the BGP VPWS with the following differences

- As stated in Section 3 of RFC 6624, there are two modifications of messages when compared to RFC 4761.
  - → The Encaps Types supported in the associated extended community.
  - → The addition of a circuit status vector sub-TLV at the end of the VPWS NLRI.
- The Control Flags and VPLS preference in the associated extended community are based on draft-ietf-l2vpn-vpls-multihoming-03.

Figure 47 displays the format of the BGP VPWS update extended community.:

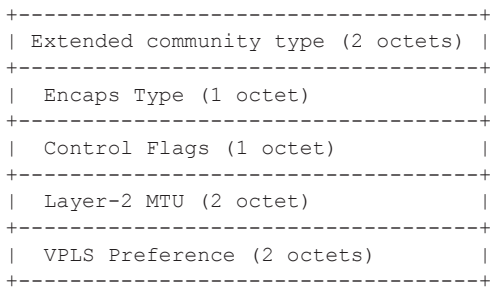


Figure 47: BGP VPWS Update Extended Community Format

- Extended community type — The value allocated by IANA for this attribute is 0x800A
- Encaps Type — Encapsulation type, identifies the type of pseudowire encapsulation. Ethernet VLAN (4) and Ethernet Raw mode (5), as described in RFC 4448, are the only values supported. If there is a mismatch between the Encaps Type signaled and the one received, the pseudowire is created but with the oper state down.
- Control Flags — Control information regarding the pseudowires, see below for details.
- Layer-2 MTU is the Maximum Transmission Unit to be used on the pseudowires. If the received Layer-2 MTU is zero no MTU check is performed and the related pseudowire is established. If there is a mismatch between the local service-mtu and the received Layer-2 MTU the pseudowire is created with the oper state down and a MTU/Parameter mismatch indication.
- VPLS preference – VPLS preference has a default value of zero for BGP-VPWS updates sent by the system, indicating that it is not in use. If the site-preference is configured, its value is used for the VPLS preference and is also used in the local designated forwarder election. On receipt of a BGP VPWS update containing a non-zero value, it will be used to determine to which system the pseudowire is established as part of the VPWS update process tie-breaking rules. The BGP local preference of the BGP VPWS update sent by the system is set to the same value as the VPLS preference if the latter is non-zero, as required by the draft (as long as the D bit in the extended community is not set to 1). Consequently, attempts to change the BGP local preference when exporting a BGP VPWS update with a non-zero VPLS preference will be ignored. This prevents the updates being treated as malformed by the receiver of the update.

The control flags are described below:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|D|A|F|Z|Z|Z|C|S| (Z = MUST Be Zero)
+-+-+-+-+-+-+-+-+
```

The following bits in the Control Flags are defined:

D — Access circuit down indicator from draft-kothari-l2vpn-auto-site-id-01. D is 1 if all access circuits are down, otherwise D is 0.

A — Automatic site id allocation, which is not supported. This is ignored on receipt and set to 0 on sending.

F — MAC flush indicator. This is not supported as it relates to a VPLS service. This is set to 0 and ignored on receipt.

C — Presence of a control word. Control word usage is supported. When this is set to 1, packets will be send and are expected to be received, with a control word. When this is set to 0, packets will be send and are expected to be received, without a control word. This is the default.

S — Sequenced delivery. Sequenced delivery is not supported. This is set to 0 on sending (no sequenced delivery) and if a non-zero value is received (indicating sequenced delivery required) the pseudowire will not be created.

The BGP VPWS NLRI is based on that defined for BGP VPLS but is extended with a circuit status vector, as shown in Figure 48.

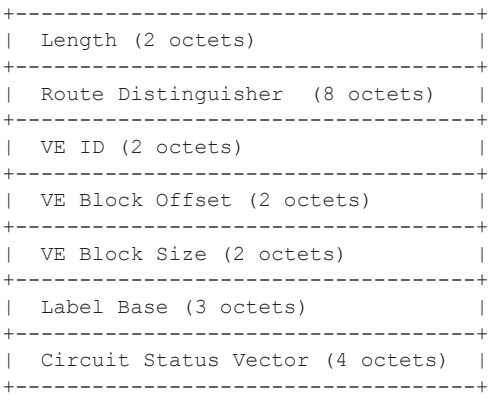


Figure 48: BGP VPWS NLRI

The VE ID value is configured within each BGP VPWS service, the label base is chosen by the system and the VE block offset corresponds to the remote VE ID as a VE block size of 1 is always used.

The circuit status vector is encoded as a TLV as shown in Figure 49.

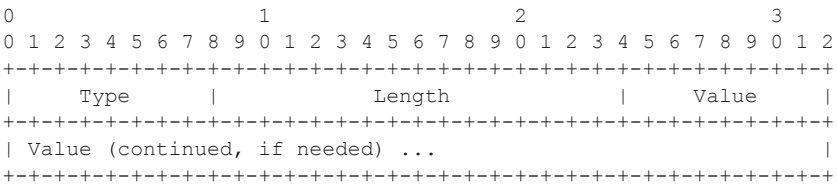


Figure 49: BGP VPWS NLRI TLV Extension Format

TLV Type	Description
1	Circuit Status Vector

Figure 50: Circuit Status Vector TLV Type

The circuit status vector is used to indicate the status of both the SAP/GRE tunnel and the status of the spoke-SDP within the local service. As the VE block size used is 1, the most significant bit in the circuit status vector TLV value will be set to 1 if either the SAP/GRE tunnel or spoke-SDP is down, otherwise it will be set to 0. On receiving a circuit status vector, only the most significant byte of the CSV is examined for designated forwarder selection purposes.

If a circuit status vector length field of greater than 32 is received, the update will be ignored and not reflected to BGP neighbors. If the length field of greater than 800, a notification message will be sent and the BGP session will restart. Also, BGP VPWS services support a single access circuit, consequently only the most significant bit of the CSV is examined on receipt.

A pseudowire will be established when a BGP VPWS update is received which matches the service configuration, specifically the configured route-targets and remote VE ID. If multiple matching updates are received, the system to which the pseudowire is established is determined by the tie-breaking rules, as described in draft-ietf-l2vpn-vpls-multihoming-03.

Traffic will be sent on the active pseudowire connected to the remote designated forwarder. It can be received on either the active or standby pseudowire, though no traffic should be received on the standby pseudowire as the SAP/GRE tunnel on the non-designated forwarder should be blocked.

BGP VPWS Configuration Procedure

In addition to configuring the associated BGP and MPLS infrastructure, the provisioning of a BGP VPWS service requires:

- Configure BGP Route Distinguisher, Route Target
  - → Updates are accepted into the service only if they contain the configured import route-target
- Configure a binding to the pseudowire template
  - → Multiple pseudowire template bindings can be configure with their associated route-targets used to control which is applied
- Configure the SAP or static GRE tunnel.
- Configure the name of the local VE and its associate VE ID
- Configure the name of the remote VE and its associated VE ID
- For a dual-homed PE
  - → Enable the site
  - → Configure the site with non-zero site-preference
- For a remote PE
  - → Up to two remove VE names and associated VE IDs can be configured
- Enable BGP VPWS

Use of Pseudowire Template for BGP VPWS

The pseudowire template concept used for BGP AD is re-used for BGP VPWS to dynamically instantiate pseudowire (SDP-bindings) and the related SDP (provisioned or automatically instantiated).

The settings for the L2-Info extended community in the BGP Update sent by the system are derived from the pseudowire-template attributes. The following rules apply:

- If multiple pseudowire-template-bindings (with or without import-rt) are specified for the VPWS instance, the first (numerically lowest id) pseudowire-template entry will be used.
- Both Ethernet VLAN and Ethernet Raw Mode encaps types are supported; these are selected by configuring the vc-type in the pseudowire template to be either vlan or ether, respectively. The default is ether.
  - → The same value must be used by the remote BGP VPWS instance to ensure the related pseudowire will come up
- Layer 2 MTU – derived from service vpls service-mtu parameter.
  - → The same value must be used by the remote BGP VPWS instance to ensure the related pseudowire will come up.
- Control Flag C – can be 0 or 1, depending on the setting of the controlword parameter in the pw-template 0.
- Control Flag S – always 0.

On reception the values of the parameters in the L2-Info extended community of the BGP update are compared with the settings from the corresponding pseudowire-template. The following steps are used to determine the local pseudowire-template:

- The route-target values are matched to determine the pseudowire-template.
- If no matches are found from the previous step, the first (numerically lowest id) pw-template-binding configured without an import-rt is used.
- If the values used for encaps type or Layer 2 MTU do not match the pseudowire is created but with the oper state down.
  - → In order to interoperate with existing implementations if the received MTU value = 0, then MTU negotiation does not take place; the related pseudowire is setup ignoring the MTU.
- If the values of the S flag is not zero the pseudowire is not created.

The following pseudowire template parameters are supported when applied within a BGP VPWS service, the remainder are ignored:

```
configure service pw-template policy-id [use-provisioned-sdp] [create]
     accounting-policy acct-policy-id
     no accounting-policy
     [no] collect-stats
     [no] controlword
     egress
          filter ipv6 ipv6-filter-id
          filter ip ip-filter-id
          filter mac mac-filter-id
          no filter [ip ip-filter-id] [mac mac-filter-id] [ipv6 ipv6-filter-id]
          qos network-policy-id port-redirect-group queue-group-name instance instance-id
          no qos [network-policy-id]
     [no] force-vlan-vc-forwarding
     hash-label [signal-capability]
     no hash-label
     ingress
          filter ipv6 ipv6-filter-id
          filter ip ip-filter-id
          filter mac mac-filter-id
          no filter [ip ip-filter-id] [mac mac-filter-id] [ipv6 ipv6-filter-id]
          qos network-policy-id fp-redirect-group queue-group-name instance instance-id
          no qos [network-policy-id]
     [no] sdp-exclude
     [no] sdp-include
     vc-type {ether|vlan}
     vlan-vc-tag vlan-id
     no vlan-vc-tag
```

The **use-provisioned-sdp** command is permitted when creating the pseudowire template if a pre-provisioned SDP is to be used. Pre-provisioned SDPs must be configured whenever RSVP or BGP signaled transport tunnels are used.

The **tools perform** command can be used similarly as for BGP-AD to force the application of changes in pseudowire-template using the format described below:

```
tools perform service [id service-id] eval-pw-template policy-id [allow-service-impact]
```

Use of Endpoint for BGP VPWS

An Endpoint is required on a remote PE connecting to two dual-homed PEs to associate the active/standby pseudowires with the Epipe service. An endpoint is automatically created within the Epipe service such that active/standby pseudowires are associated with that endpoint. The creation of the endpoint occurs when bgp-vpws is enabled (and deleted when it is disabled) and so will exist in both a single and dual homed scenario (this simplifies converting a single homed service to a dual-homed service). The naming convention used is _tmnx_BgpVpws-x, where x is the service identifier. The automatically created endpoint has the default parameter values, although all are ignored in a BGP-VPWS service with the description field being defined by the system.

Note that the command:

```
tools perform service id <service-id> endpoint <endpoint-name> force-switchover
```

will have no affect on an automatically created VPWS endpoint.

VLL Service Considerations

This section describes the general , 7750 SR, and service features and any special capabilities or considerations as they relate to VLL services.

SDPs

The most basic SDPs must have the following:

- A locally unique SDP identification (ID) number.
- The system IP address of the originating and far-end routers.
- An SDP encapsulation type, either GRE or MPLS.

The most basic Apipe and Fpipe SDP configurations must have the following:

- A locally unique SDP identification (ID) number and vc-id.

SDP Statistics for VPLS and VLL Services

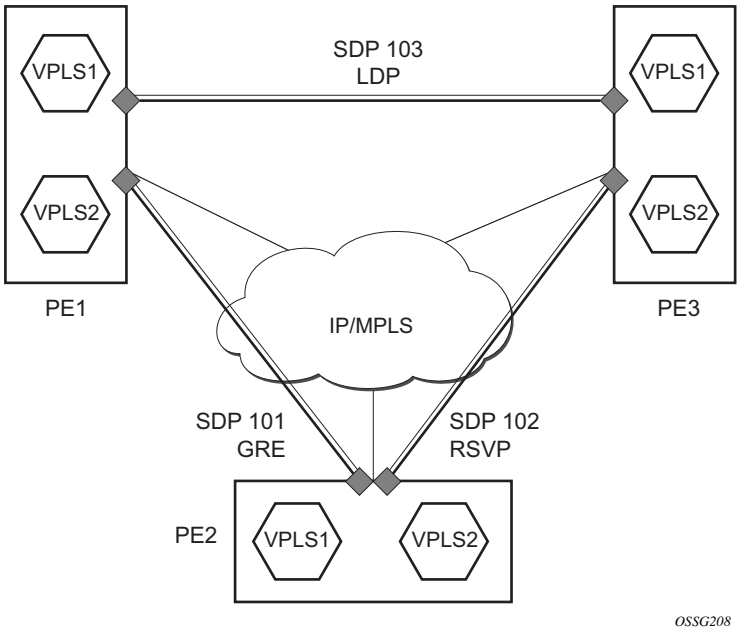


Figure 51: SDP Statistics for VPLS and VLL Services

The simple three-node network described in Figure 51 shows two MPLS SDPs and one GRE SDP defined between the nodes. These SDPs connect VPLS1 and VPLS2 instances that are defined in the three nodes. With this feature the operator will have local CLI based as well as SNMP based statistics collection for each VC used in the SDPs. This will allow for traffic management of tunnel usage by the different services and with aggregation the total tunnel usage.

SAP Encapsulations and Pseudowire Types

The Epipe service is designed to carry Ethernet frame payloads, so it can provide connectivity between any two SAPs that pass Ethernet frames. The following SAP encapsulations are supported on the , 7750 SR, and Epipe service:

- Ethernet null
- Ethernet dot1q
- QinQ
- SONET/SDH BCP-null
- SONET/SDH BCP-dot1q
- ATM VC with RFC 2684 Ethernet-bridged encapsulation (see Ethernet Interworking VLL on page 51)
- FR VC with RFC 2427 Ethernet-bridged encapsulation (see Ethernet Interworking VLL on page 51)

Note that while different encapsulation types can be used, encapsulation mismatching can occur if the encapsulation behavior is not understood by connecting devices and are unable to send and receive the expected traffic. For example if the encapsulation type on one side of the Epipe is dot1q and the other is null, tagged traffic received on the null SAP will be double tagged when it is transmitted out of the Dot1q SAP.

ATM VLLs can be configured with both endpoints (SAPs) on the same router or with the two endpoints on different 7750 SRs. In the latter case, Pseudowire Emulation Edge-to-Edge (PWE3) signalling is used to establish a pseudowire between the devices allowing ATM traffic to be tunneled through an MPLS or GRE network:

Two pseudowire encapsulation modes, i.e., SDP vc-type, are available:

- PWE3 N-to-1 Cell Mode Encapsulation
- PWE3 AAL5 SDU Mode Encapsulation

The endpoints of Frame Relay VLLs must be Data-Link Connection Identifiers (DLCIs) on any port that supports Frame Relay. The pseudowire encapsulation, or SDP vc-type, supported is the 1-to-1 Frame Relay encapsulation mode.

PWE3 N-to-1 Cell Mode

The endpoints of an N-to-1 mode VLL can be:

- ATM VCs — VPI/VCI translation is supported (i.e., the VPI/VCI at each endpoint does not need to be the same).
- ATM VPs — VPI translation is supported (i.e., the VPI at each endpoint need not be the same, but the original VCI will be maintained).
- ATM VTs (a VP range) — No VPI translation is supported (i.e., the VPI/VCI of each cell is maintained across the network).
- ATM ports — No translation is supported (i.e., the VPI/VCI of each cell is maintained across the network).

For N-to-1 mode VLLs, cell concatenation is supported. Cells will be packed on ingress to the VLL and unpacked on egress. As cells are being packed, the concatenation process may be terminated by:

- Reaching a maximum number of cells per packet.
- Expiry of a timer.
- (Optionally) change of the CLP bit.
- (Optionally) change of the PTI bits indicating end of AAL5 packet.

In N-to-1 mode, OAM cells are transported through the VLL as any other cell. The PTI and CLP bits are untouched, even if VPI/VCI translation is carried out.

PWE3 AAL5 SDU Mode

The endpoints of an AAL5 SDU mode VLL must be ATM VCs specified by port/vpi/vci. VPI/VCI translation is supported. The endpoint can also be a FR VC, specified by port/dlci. In this case FRF.5 FR-ATM network interworking is performed between the ATM VC SAP or the SDP and the FR VC SAP.

In SDU mode, the mandatory PWE3 control word is supported. This allows the ATM VLL to transport OAM cells along with SDU frames, using the “T” bit to distinguish between them, to recover the original SDU length, and to carry CLP, EFCI and UU information.

QoS Policies

When applied to , 7750 SR, or Epipe, Apipe, and Fpipe services, service ingress QoS policies only create the unicast queues defined in the policy. The multipoint queues are not created on the service.

With Epipe, Apipe, and Fpipe services, egress QoS policies function as with other services where the class-based queues are created as defined in the policy. Note that both Layer 2 or Layer 3 criteria can be used in the QoS policies for traffic classification in a service. QoS policies on Apipes cannot perform any classification and on Fpipes Layer 3 (IP) classification is performed.

Filter Policies

, 7750 SR, and Epipe, Fpipe, and Ipipe services can have a single filter policy associated on both ingress and egress. Both MAC and IP filter policies can be used on Epipe services.

Filters cannot be configured on 7750 SR Apipe service SAPs.

MAC Resources

Epipe services are point-to-point layer 2 VPNs capable of carrying any Ethernet payloads. Although an Epipe is a Layer 2 service, the , 7750 SR, and Epipe implementation does not perform any MAC learning on the service, so Epipe services do not consume any MAC hardware resources.