Configuring a VPLS Service with CLI

This section provides information to configure VPLS services using the command line interface.

Topics in this section include:

Basic Configuration

The following fields require specific input (there are no defaults) to configure a basic VPLS service:

- Customer ID (refer to Configuring Customers on page 103)
- For a local service, configure two SAPs, specifying local access ports and encapsulation values.
- For a distributed service, configure a SAP and an SDP for each far-end node.

The following example displays a sample configuration of a local VPLS service on ALA-1.

```
*A:ALA-1>config>service>vpls# info
----------------------------------------------
...
        vpls 9001 customer 6 create
            description "Local VPLS"
            stp
                shutdown
            exit
            sap 1/2/2:0 create
                description "SAP for local service"
            exit
            sap 1/1/5:0 create
                description "SAP for local service"
            exit
            no shutdown
----------------------------------------------
*A:ALA-1>config>service>vpls#
```

The following example displays a sample configuration of a distributed VPLS service between ALA-1, ALA-2, and ALA-3.

```
*A:ALA-1>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 create
            shutdown
            description "This is a distributed VPLS."
            def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/1/5:16 create
                description "VPLS SAP"
            exit
            spoke-sdp 2:22 create
            exit
            mesh-sdp 7:750 create
            exit
        exit
...
----------------------------------------------
*A:ALA-1>config>service#
```

```
*A:ALA-2>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 create
            description "This is a distributed VPLS."
            def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/1/5:16 create
                description "VPLS SAP"
            exit
            spoke-sdp 2:22 create
            exit
            mesh-sdp 8:750 create
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-2>config>service#

*A:ALA-3>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 create
            description "This is a distributed VPLS."
           def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/1/3:33 create
                description "VPLS SAP"
            exit
            spoke-sdp 2:22 create
            exit
            mesh-sdp 8:750 create
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-3>config>service#
```

Common Configuration Tasks

This section provides a brief overview of the tasks that must be performed to configure both local and distributed VPLS services and provides the CLI commands.

For egress multicast groups (optional):

1. Define egress multicast group name(s)
2. Specify the destinations per pass
3. Define SAP common requirements

For VPLS services:

1. Associate VPLS service with a customer ID
2. Define SAPs:
   - Select node(s) and port(s)
   - Optional — Select QoS policies other than the default (configured in `config>qos` context)
   - Optional — Select filter policies (configured in `config>filter` context)
   - Optional — Select accounting policy (configured in `config>log` context)

   — Optional — Specify SAP egress multicast-group name
3. Associate SDPs for (distributed services)
4. Modify STP default parameters (optional) (see VPLS and Spanning Tree Protocol on page 622)
5. Enable service

Configuring VPLS Components

Use the CLI syntax displayed below to configure the following entities:

Configuring Egress Multicast Groups

Use the following CLI syntax to configure egress multicast groups:

CLI Syntax:
```
config>service# egress-multicast-group group-name
             description description-string
             dest-chain-limit [destinations per pass]
             sap-common-requirements
                dot1q-etype 0x0600..0xffff
                egress-filter [ip ip-filter-id]
                egress-filter [ipv6 ipv6-filter-id]
                egress-filter [mac mac-filter-id]
                qinq-etype [0x0600..0xffff]
                qinq-fixed-tag-value tag-value
```

The following example displays an egress multicast group configuration:

```
A:ALA-48>config>service>egress-multicast-group# info
----------------------------------------------
               dest-chain-limit 10
               sap-common-requirements
                   dot1q-etype 0x060e
                   egress-filter ip 10
               exit
----------------------------------------------
A:ALA-48>config>service>egress-multicast-group#
```

Creating a VPLS Service

Use the following CLI syntax to create a VPLS service:

CLI Syntax: config>service# vpls *service-id* [customer *customer-id*] [vpn *vpn-id*] [m-vpls] [b-vpls | i-vpls] [create]
description *description-string*
no shutdown

The following example displays a VPLS configuration:

```
*A:ALA-1>config>service>vpls# info
----------------------------------------------
...
        vpls 9000 customer 6 create
            description "This is a distributed VPLS."
            def-mesh-vc-id 750
            stp
                shutdown
            exit
        exit
...
----------------------------------------------
*A:ALA-1>config>service>vpls#
```

Enabling Multiple MAC Registration Protocol (MMRP)

Once MMRP is enabled in the B-VPLS, it advertises the presence of the I-VPLS instances associated with this B-VPLS.

The following example displays a configuration with MMRP enabled.

```
*A:PE-B>config>service# info
----------------------------------------------
        vpls 11 customer 1 vpn 11 i-vpls create
            backbone-vpls 100:11
            exit
            stp
                shutdown
            exit
            sap 1/5/1:11 create
            exit
            sap 1/5/1:12 create
            exit
            no shutdown
        exit
        vpls 100 customer 1 vpn 100 b-vpls create
            service-mtu 2000
            stp
                shutdown
            exit
            mrp
                flood-time 10
                no shutdown
            exit
            sap 1/5/1:100 create
            exit
            spoke-sdp 3101:100 create
            exit
            spoke-sdp 3201:100 create
            exit
            no shutdown
        exit
----------------------------------------------
*A:PE-B>config>service#
```

Since I-VPLS 11 is associated with B-VPLS 100, MMRP advertises the group B-MAC 01:1e:83:00:00:0b) associated with I-VPLS 11 through a declaration on all the B-SAPs and B-SDPs. If the remote node also declares an I-VPLS 11 associated to its B-VPLS 10, then this results in a registration for the group B-MAC. This also creates the MMRP multicast tree (MFIB entries). In this case, sdp 3201:100 is connected to a remote node that declares the group B-MAC.

The following show commands display the current MMRP information for this scenario:

```
*A:PE-C# show service id 100 mrp
-------------------------------------------------------------------------------
MRP Information
-------------------------------------------------------------------------------
Admin State        : Up                Failed Register Cnt: 0
Max Attributes     : 1023              Attribute Count    : 1
```

```
Attr High Watermark: 95%                Attr Low Watermark : 90%
Flood Time         : 10
-------------------------------------------------------------------------------
*A:PE-C# show service id 100 mmrp mac
-------------------------------------------------------------------------------
SAP/SDP                                 MAC Address       Registered  Declared
-------------------------------------------------------------------------------
sap:1/5/1:100                           01:1e:83:00:00:0b No          Yes
sdp:3101:100                            01:1e:83:00:00:0b No          Yes
sdp:3201:100                            01:1e:83:00:00:0b Yes         Yes
-------------------------------------------------------------------------------


*A:PE-C# show service id 100 sdp 3201:100 mrp
-------------------------------------------------------------------------------
Sdp Id 3201:100 MRP Information
-------------------------------------------------------------------------------
Join Time          : 0.2 secs                Leave Time        : 3.0 secs
Leave All Time     : 10.0 secs               Periodic Time     : 1.0 secs
Periodic Enabled   : false
Rx Pdus            : 7                       Tx Pdus           : 23
Dropped Pdus       : 0
Rx New Event       : 0                       Rx Join-In Event  : 6
Rx In Event        : 0                       Rx Join Empty Evt : 1
Rx Empty Event     : 0                       Rx Leave Event    : 0
Tx New Event       : 0                       Tx Join-In Event  : 4
Tx In Event        : 0                       Tx Join Empty Evt : 19
Tx Empty Event     : 0                       Tx Leave Event    : 0
-------------------------------------------------------------------------------
SDP MMRP Information
-------------------------------------------------------------------------------
MAC Address       Registered        Declared
-------------------------------------------------------------------------------
01:1e:83:00:00:0b Yes               Yes
-------------------------------------------------------------------------------
Number of MACs=1 Registered=1 Declared=1
-------------------------------------------------------------------------------
*A:PE-C#


*A:PE-C#  show service id 100 mfib
===============================================================================
Multicast FIB, Service 100
===============================================================================
Source Address  Group Address         Sap/Sdp Id               Svc Id   Fwd/Blk
-------------------------------------------------------------------------------
*               01:1E:83:00:00:0B     sdp:3201:100             Local    Fwd
-------------------------------------------------------------------------------
Number of entries: 1
===============================================================================
*A:PE-C#
```

Enabling MAC Move

The **mac-move** feature is useful to protect against undetected loops in your VPLS topology as well as the presence of duplicate MACs in a VPLS service. For example, if two clients in the VPLS have the same MAC address, the VPLS will experience a high re-learn rate for the MAC and will shut down the SAP or spoke SDP when the threshold is exceeded.

Use the following CLI syntax to configure **mac-move** parameters.

CLI Syntax: config>service# vpls *service-id* [customer *customer-id*] [vpn *vpn-id*] [m-vpls]

```
            mac-move
               primary-ports
                  spoke-sdp
                  cumulative-factor
               exit
               secondary-ports
                  spoke-sdp
                  sap
               exit
               move-frequency frequency
               retry-timeout timeout
               no shutdown
```

The following example displays a **mac-move** configuration:

```
*A:ALA-2009>config>service>vpls>mac-move# show service id 500 mac-move
===============================================================================
Service Mac Move Information
===============================================================================
Service Id        : 500                 Mac Move          : Enabled
Primary Factor    : 4                   Secondary Factor  : 2
Mac Move Rate     : 2                   Mac Move Timeout  : 10
Mac Move Retries  : 3
-------------------------------------------------------------------------------
SAP Mac Move Information: 2/1/3:501
-------------------------------------------------------------------------------
Admin State       : Up                  Oper State        : Down
Flags              : RelearnLimitExceeded
Time to come up   : 1 seconds           Retries Left      : 1
Mac Move          : Blockable           Blockable Level   : Tertiary
-------------------------------------------------------------------------------
SAP Mac Move Information: 2/1/3:502
-------------------------------------------------------------------------------
Admin State       : Up                  Oper State        : Up
Flags              : None
Time to RetryReset: 267 seconds         Retries Left      : none
Mac Move          : Blockable           Blockable Level   : Tertiary
-------------------------------------------------------------------------------
SDP Mac Move Information: 21:501
-------------------------------------------------------------------------------
Admin State       : Up                  Oper State        : Up
Flags              : None
Time to RetryReset: never               Retries Left      : 3
Mac Move          : Blockable           Blockable Level   : Secondary
```

```
-------------------------------------------------------------------------------
SDP Mac Move Information: 21:502
-------------------------------------------------------------------------------
Admin State       : Up                  Oper State        : Down
Flags              : RelearnLimitExceeded
Time to come up   : never               Retries Left      : none
Mac Move          : Blockable           Blockable Level   : Tertiary
===============================================================================
*A:*A:ALA-2009>config>service>vpls>mac-move#
```

Configuring STP Bridge Parameters in a VPLS

Modifying some of the Spanning Tree Protocol parameters allows the operator to balance STP between resiliency and speed of convergence extremes. Modifying particular parameters, mentioned below, must be done in the constraints of the following two formulae:

2 x (*Bridge_Forward_Delay* - 1.0 seconds) >= *Bridge_Max_Age*
Bridge_Max_Age >= 2 x (*Bridge_Hello0_Time* + 1.0 seconds)

The following STP parameters can be modified at VPLS level:

- Bridge STP Admin State on page 744
- Mode on page 745
- Bridge Priority on page 745
- Max Age on page 746
- Forward Delay on page 746
- Hello Time on page 747
- MST Instances on page 748
- MST Max Hops on page 748
- MST Name on page 748
- MST Revision on page 748

STP always uses the locally configured values for the first three parameters (Admin State, Mode and Priority).

For the parameters Max Age, Forward Delay, Hello Time and Hold Count, the locally configured values are only used when this bridge has been elected root bridge in the STP domain, otherwise the values received from the root bridge are used. The exception to this rule is: when STP is running in RSTP mode, the Hello Time is always taken from the locally configured parameter. The other parameters are only used when running mode MSTP.

Bridge STP Admin State

The administrative state of STP at the VPLS level is controlled by the shutdown command.

When STP on the VPLS is administratively disabled, any BPDUs are forwarded transparently through the 7750 SR. When STP on the VPLS is administratively enabled, but the administrative state of a SAP or spoke SDP is down, BPDUs received on such a SAP or spoke SDP are discarded.

CLI Syntax: `config>service>vpls` *`service-id`*`# stp`
`no shutdown`

Mode

To be compatible with the different iterations of the IEEE 802.1D standard, the 7750 SR supports several variants of the Spanning Tree protocol:

- `rstp` — Rapid Spanning Tree Protocol (RSTP) compliant with IEEE 802.1D-2004 - default mode.
- `dot1w` — Compliant with IEEE 802.1w.
- `comp-dot1w` — Operation as in RSTP but backwards compatible with IEEE 802.1w (this mode was introduced for interoperability with some MTU types).
- `mstp` — Compliant with the Multiple Spanning Tree Protocol specified in IEEE 802.1Q REV/D5.0-09/2005. This mode of operation is only supported in an mVPLS.
- `pmstp` — Compliant with the Multiple Spanning Tree Protocol specified in IEEE 802.1Q REV/D3.0-04/2005 but with some changes to make it backwards compatible to 802.1Q 2003 edition and IEEE 802.1w.

See section Spanning Tree Operating Modes on page 622 for details on these modes.

CLI Syntax: `config>service>vpls` *`service-id`*`# stp`
`mode {rstp | comp-dot1w | dot1w | mstp}`
Default: `rstp`

Bridge Priority

The **bridge-priority** command is used to populate the priority portion of the bridge ID field within outbound BPDUs (the most significant 4 bits of the bridge ID). It is also used as part of the decision process when determining the best BPDU between messages received and sent.When running MSTP, this is the bridge priority used for the CIST.

All values will be truncated to multiples of 4096, conforming with IEEE 802.1t and 802.1D-2004.

CLI Syntax: `config>service>vpls` *`service-id`*`# stp`
`priority` *`bridge-priority`*
Range: 1 to 65535
Default: 32768
Restore Default: `no priority`

Max Age

The **max-age** command indicates how many hops a BPDU can traverse the network starting from the root bridge. The message age field in a BPDU transmitted by the root bridge is initialized to 0. Each other bridge will take the message_age value from BPDUs received on their root port and increment this value by 1. The message_age thus reflects the distance from the root bridge. BPDUs with a message age exceeding max-age are ignored.

STP uses the max-age value configured in the root bridge. This value is propagated to the other bridges by the BPDUs.
The default value of **max-age** is 20. This parameter can be modified within a range of 6 to 40, limited by the standard STP parameter interaction formulae.

CLI Syntax: `config>service>vpls` *`service-id`*`# stp`
`max-age` *`max-info-age`*
Range: 6 to 40 seconds
Default: 20 seconds
Restore Default: `no max-age`

Forward Delay

RSTP, as defined in the IEEE 802.1D-2004 standards, will normally transition to the forwarding state by a handshaking mechanism (rapid transition), without any waiting times. If handshaking fails (e.g. on shared links, see below), the system falls back to the timer-based mechanism defined in the original STP (802.1D-1998) standard.

A shared link is a link with more than two Ethernet bridges (for example, a shared 10/100BaseT segment). The `port-type` command is used to configure a link as point-to-point or shared (see section SAP Link Type on page 756).

For timer-based transitions, the 802.1D-2004 standard defines an internal variable forward-delay, which is used in calculating the default number of seconds that a SAP or spoke SDP spends in the discarding and learning states when transitioning to the forwarding state. The value of the forward-delay variable depends on the STP operating mode of the VPLS instance:

- in `rstp` mode, but only when the SAP or spoke SDP has not fallen back to legacy STP operation, the value configured by the **hello-time** command is used;
- in all other situations, the value configured by the **forward-delay** command is used.

CLI Syntax: `config>service>vpls` *`service-id`*`# stp`
`forward-delay` *`seconds`*
Range: 4 to 30 seconds
Default: 15 seconds
Restore Default: `no forward-delay`

Hello Time

The **hello-time** command configures the Spanning Tree Protocol (STP) hello time for the Virtual Private LAN Service (VPLS) STP instance.

The *seconds* parameter defines the default timer value that controls the sending interval between BPDU configuration messages by this bridge, on ports where this bridge assumes the designated role.

The active hello time for the spanning tree is determined by the root bridge (except when the STP is running in RSTP mode, then the hello time is always taken from the locally configured parameter).

The configured hello-time value can also be used to calculate the bridge forward delay, see Forward Delay on page 746.

CLI Syntax: `config>service>vpls service-id# stp`
`hello-time hello-time`
Range: 1 to 10 seconds
Default: 2 seconds
Restore Default: `no hello-time`

Hold Count

The **hold-count** command configures the peak number of BPDUs that can be transmitted in a period of one second.

CLI Syntax: `config>service>vpls service-id# stp`
`hold-count count-value`
Range: 1 to 10
Default: 6
Restore Default: `no hold-count`

MST Instances

You can create up to 15 MST-instances. They can range from 1 to 4094. By changing path-cost and priorities, you can make sure that each instance will form it's own tree within the region, thus making sure different VLANs follow different paths.

You can assign non overlapping VLAN ranges to each instance. VLANs that are not assigned to an instance are implicitly assumed to be in instance 0, which is also called the CIST. This CIST cannot be deleted or created.

The parameter that can be defined per instance are mst-priority and vlan-range.

- mst-priority — The bridge-priority for this specific mst-instance. It follows the same rules as bridge-priority. For the CIST, the bridge-priority is used.
- vlan-range — The VLANs are mapped to this specific mst-instance. If no VLAN-ranges are defined in any mst-instances, then all VLANs are mapped to the CIST.

MST Max Hops

The mst-max-hops command defines the maximum number of hops the BPDU can traverse inside the region. Outside the region max-age is used.

MST Name

The MST name defines the name that the operator gives to a region. Together with MST revision and the VLAN to MST-instance mapping, it forms the MST configuration identifier. Two bridges that have the same MST configuration identifier form a region if they exchange BPDUs.

MST Revision

The MST revision together with MST-name and VLAN to MST-instance mapping define the MST configuration identifier. Two bridges that have the same MST configuration identifier form a region if they exchange BPDUs.

Configuring GSMP Parameters

The following parameters must be configured in order for GSMP to function:

- One or more GSMP sessions
- One or more ANCP policies
- For basic subscriber management only, ANCP static maps
- For enhanced subscriber management only, associate subscriber profiles with ANCP policies.

Use the following CLI syntax to configure GSMP parameters.

CLI Syntax:

```
config>service>vpls# gsmp
          group name [create]
             ancp
                dynamic-topology-discover
                oam
             description description-string
             hold-multiplier multiplier
             keepalive seconds
             neighbor ip-address [create]
                description v
                local-address ip-address
                priority-marking dscp dscp-name
                priority-marking prec ip-prec-value
                [no] shutdown
             [no] shutdown
          [no] shutdown
```

This example displays a GSMP group configuration.

```
A:ALA-48>config>service>vpls>gsmp# info
----------------------------------------------
               group "group1" create
                   description "test group config"
                   neighbor 10.10.10.104 create
                       description "neighbor1 config"
                       local-address 10.10.10.103
                       no shutdown
                   exit
                   no shutdown
               exit
               no shutdown
----------------------------------------------
A:ALA-48>config>service>vpls>gsmp#
```

Configuring a VPLS SAP

A default QoS policy is applied to each ingress and egress SAP. Additional QoS policies can be configured in the **config>qos** context. There are no default filter policies. Filter policies are configured in the **config>filter** context and must be explicitly applied to a SAP.
Use the following CLI syntax to create:

- Local VPLS SAPs on page 750
- Distributed VPLS SAPs on page 751

Local VPLS SAPs

To configure a local VPLS service, enter the **sap** *sap-id* command twice with different port IDs in the same service configuration.

The following example displays a local VPLS configuration:

```
*A:ALA-1>config>service# info
----------------------------------------------
...
        vpls 90001 customer 6 create
            description "Local VPLS"
            stp
                shutdown
            exit
            sap 1/2/2:0 create
                description "SAP for local service"
            exit
            sap 1/1/5:0 create
                description "SAP for local service"
            exit
            no shutdown
        exit
----------------------------------------------
*A:ALA-1>config>service#
*A:ALA-1>config>service# info
----------------------------------------------
       vpls 1150 customer 1 create
            fdb-table-size 1000
            fdb-table-low-wmark 5
            fdb-table-high-wmark 80
            local-age 60
            stp
                shutdown
            exit
            sap 1/1/1:1155 create
            exit
            sap 1/1/2:1150 create
            exit
            no shutdown
        exit
----------------------------------------------
*A:ALA-1>config>service#
```

Distributed VPLS SAPs

To configure a distributed VPLS service, you must configure service entities on originating and far-end nodes. You must use the same service ID on all ends (for example, create a VPLS service ID 9000 on ALA-1, ALA-2, and ALA-3). A distributed VPLS consists of a SAP on each participating node and an SDP bound to each participating node.

For SDP configuration information, see Configuring an SDP on page 107. For SDP binding information, see Configuring SDP Bindings on page 764.

The following example displays a configuration of VPLS SAPs configured for ALA-1, ALA-2, and ALA-3.

```
*A:ALA-1>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 vpn 750 create
            description "Distributed VPLS services."
           def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/2/5:0 create
                description "VPLS SAP"
                multi-service-site "West"
            exit
        exit
...
----------------------------------------------
*A:ALA-1>config>service#


*A:ALA-2>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 vpn 750 create
            description "Distributed VPLS services."
           def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/1/2:22 create
                description "VPLS SAP"
                multi-service-site "West"
            exit
        exit
...
----------------------------------------------
*A:ALA-2>config>service#


*A:ALA-3>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 vpn 750 create
            description "Distributed VPLS services."
           def-mesh-vc-id 750
```

```
            stp
                shutdown
            exit
            sap 1/1/3:33 create
                description "VPLS SAP"
                multi-service-site "West"
            exit
        exit
...
-------------------------------------------------
*A:ALA-3>config>service#
```

Configuring SAP-Specific STP Parameters

When a VPLS has STP enabled, each SAP within the VPLS has STP enabled by default. The operation of STP on each SAP is governed by:

SAP STP Administrative State

The administrative state of STP within a SAP controls how BPDUs are transmitted and handled when received. The allowable states are:

- SAP Admin Up

  The default administrative state is *up* for STP on a SAP. BPDUs are handled in the normal STP manner on a SAP that is administratively up.

- SAP Admin Down

  An administratively down state allows a service provider to prevent a SAP from becoming operationally blocked. BPDUs will not originate out the SAP towards the customer.

  If STP is enabled on VPLS level, but disabled on the SAP, received BPDUs are discarded. Discarding the incoming BPDUs allows STP to continue to operate normally within the VPLS service while ignoring the down SAP. The specified SAP will always be in an operationally forwarding state.

  NOTE: The administratively down state allows a loop to form within the VPLS.

CLI Syntax: `config>service>vpls>sap>stp#`
`[no] shutdown`
Range: `shutdown` or `no shutdown`
Default: `no shutdown` (SAP admin up)

SAP Virtual Port Number

The virtual port number uniquely identifies a SAP within configuration BPDUs. The internal representation of a SAP is unique to a system and has a reference space much bigger than the 12 bits definable in a configuration BPDU. STP takes the internal representation value of a SAP and identifies it with it's own virtual port number that is unique to every other SAP defined on the VPLS. The virtual port number is assigned at the time that the SAP is added to the VPLS.

Since the order in which SAPs are added to the VPLS is not preserved between reboots of the system, the virtual port number may change between restarts of the STP instance. To achieve consistency after a reboot, the virtual port number can be specified explicitly.

CLI Syntax: `config>service>vpls>sap# stp`
`port-num` *`number`*
Range: 1 — 2047
Default: (automatically generated)
Restore Default: `no port-num`

SAP Priority

SAP priority allows a configurable "tie breaking" parameter to be associated with a SAP. When configuration BPDUs are being received, the configured SAP priority will be used in some circumstances to determine whether a SAP will be designated or blocked. These are the values used for CIST when running MSTP.

In traditional STP implementations (802.1D-1998), this field is called the port priority and has a value of 0 to 255. This field is coupled with the port number (0 to 255 also) to create a 16 bit value. In the latest STP standard (802.1D-2004) only the upper 4 bits of the port priority field are used to encode the SAP priority. The remaining 4 bits are used to extend the port ID field into a 12 bit virtual port number field. The virtual port number uniquely references a SAP within the STP instance. See SAP Virtual Port Number on page 754 for details on the virtual port number.

STP computes the actual SAP priority by taking the configured priority value and masking out the lower four bits. The result is the value that is stored in the SAP priority parameter. For example, if a value of 0 was entered, masking out the lower 4 bits would result in a parameter value of 0. If a value of 255 was entered, the result would be 240.

The default value for SAP priority is 128. This parameter can be modified within a range of 0 to 255, 0 being the highest priority. Masking causes the values actually stored and displayed to be 0 to 240, in increments of 16.

CLI Syntax: `config>service>vpls>sap>stp#`
`priority` *`stp-priority`*
Range: 0 to 255 (240 largest value, in increments of 16)
Default: 128
Restore Default: `no priority`

SAP Path Cost

The SAP path cost is used by STP to calculate the path cost to the root bridge. The path cost in BPDUs received on the root port is incremented with the configured path cost for that SAP. When BPDUs are sent out other egress SAPs, the newly calculated root path cost is used. These are the values used for CIST when running MSTP.

STP suggests that the path cost is defined as a function of the link bandwidth. Since SAPs are controlled by complex queuing dynamics, in the 7750 SR, the STP path cost is a purely static configuration.

The default value for SAP path cost is 10. This parameter can be modified within a range of 1 to 65535, 1 being the lowest cost.

CLI Syntax: `config>service>vpls>sap>stp#`
`path-cost` *`sap-path-cost`*
Range: 1 to 200000000
Default: 10
Restore Default: `no path-cost`

SAP Edge Port

The SAP `edge-port` command is used to reduce the time it takes a SAP to reach the forwarding state when the SAP is on the edge of the network, and thus has no further STP bridge to handshake with.

The `edge-port` command is used to initialize the internal OPER_EDGE variable. At any time, when OPER_EDGE is false on a SAP, the normal mechanisms are used to transition to the forwarding state (see Forward Delay on page 746). When OPER_EDGE is true, STP assumes that the remote end agrees to transition to the forwarding state without actually receiving a BPDU with an agreement flag set.

The OPER_EDGE variable will dynamically be set to false if the SAP receives BPDUs (the configured edge-port value does not change). The OPER_EDGE variable will dynamically be set to true if auto-edge is enabled and STP concludes there is no bridge behind the SAP.

When STP on the SAP is administratively disabled and re-enabled, the OPER_EDGE is re-initialized to the value configured for edge-port.

Valid values for SAP edge-port are enabled and disabled with disabled being the default.

CLI Syntax: `config>service>vpls>sap>stp#`
`[no] edge-port`
Default: `no edge-port`

SAP Auto Edge

The SAP **edge-port** command is used to instruct STP to dynamically decide whether the SAP is connected to another bridge.

If auto-edge is enabled, and STP concludes there is no bridge behind the SAP, the OPER_EDGE variable will dynamically be set to true. If auto-edge is enabled, and a BPDU is received, the OPER_EDGE variable will dynamically be set to true (see SAP Edge Port on page 755).

Valid values for SAP auto-edge are enabled and disabled with enabled being the default.

CLI Syntax: `config>service>vpls>sap>stp#`
`[no] auto-edge`
Default: `auto-edge`

SAP Link Type

The SAP **link-type** parameter instructs STP on the maximum number of bridges behind this SAP. If there is only a single bridge, transitioning to forwarding state will be based on handshaking (fast transitions). If more than two bridges are connected by a shared media, their SAPs should all be configured as shared, and timer-based transitions are used.

Valid values for SAP link-type are shared and pt-pt with pt-pt being the default.

CLI Syntax: `config>service>vpls>sap>stp#`
`link-type {pt-pt|shared}`
Default: `link-type pt-pt`
Restore Default: `no link-type`

STP SAP Operational States

The operational state of STP within a SAP controls how BPDUs are transmitted and handled when received. Defined states are:

- Operationally Disabled on page 757
- Operationally Discarding on page 757
- Operationally Learning on page 757
- Operationally Forwarding on page 758

Operationally Disabled

Operationally disabled is the normal operational state for STP on a SAP in a VPLS that has any of the following conditions:

- VPLS state administratively down
- SAP state administratively down
- SAP state operationally down

If the SAP enters the operationally up state with the STP administratively up and the SAP STP state is up, the SAP will transition to the STP SAP discarding state.

When, during normal operation, the router detects a downstream loop behind a SAP or spoke SDP, BPDUs can be received at a very high rate. To recover from this situation, STP will transition the SAP to disabled state for the configured forward-delay duration.

Operationally Discarding

A SAP in the discarding state only receives and sends BPDUs, building the local proper STP state for each SAP while not forwarding actual user traffic. The duration of the discarding state is explained in section Forward Delay on page 746.

Note: in previous versions of the STP standard, the discarding state was called a blocked state.

Operationally Learning

The learning state allows population of the MAC forwarding table before entering the forwarding state. In this state, no user traffic is forwarded.

Operationally Forwarding

Configuration BPDUs are sent out a SAP in the forwarding state. Layer 2 frames received on the SAP are source learned and destination forwarded according to the FIB. Layer 2 frames received on other forwarding interfaces and destined for the SAP are also forwarded.

SAP BPDU Encapsulation State

IEEE 802.1d (referred as Dot1d) and Cisco's per VLAN Spanning Tree (PVST) BPDU encapsulations are supported on a per SAP basis. STP is associated with a VPLS service like PVST is associated per VLAN. The main difference resides in the Ethernet and LLC framing and a type-length-value (TLV) field trailing the BPDU.

The following table shows differences between Dot1d and PVST Ethernet BPDU encapsulations based on the interface encap-type field:

Each SAP has a Read-Only operational state that shows which BPDU encapsulation is currently active on the SAP. The states are:

- Dot1d — This state specifies that the switch is currently sending IEEE 802.1d standard BPDUs. The BPDUs are tagged or non-tagged based on the encapsulation type of the egress interface and the encapsulation value defined in the SAP. A SAP defined on an interface with encapsulation type Dot1q continues in the dot1d BPDU encapsulation state until a PVST encapsulated BPDU is received in which case, the SAP will convert to the PVST encapsulation state. Each received BPDU must be properly IEEE 802.1q tagged if the interface encapsulation type is defined as Dot1q. PVST BPDUs will be silently discarded if received when the SAP is on an interface defined with encapsulation type null.
- PVST — This state specifies that the switch is currently sending proprietary encapsulated BPDUs. PVST BPDUs are only supported on Ethernet interfaces with the encapsulation type set to dot1q. The SAP continues in the PVST BPDU encapsulation state until a dot1d encapsulated BPDU is received, in which case, the SAP reverts to the dot1d encapsulation state. Each received BPDU must be properly IEEE 802.1q tagged with the encapsulation value defined for the SAP. PVST BPDUs are silently discarded if received when the SAP is on an interface defined with a null encapsulation type.

Dot1d is the initial and only SAP BPDU encapsulation state for SAPs defined on Ethernet interface with encapsulation type set to null.

Each transition between encapsulation types optionally generates an alarm that can be logged and optionally transmitted as an SNMP trap.

Configuring VPLS SAPs with Split Horizon

To configure a VPLS service with a split horizon group, add the **split-horizon-group** parameter when creating the SAP. Traffic arriving on a SAP within a split horizon group will not be copied to other SAPs in the same split horizon group.

The following example displays a VPLS configuration with split horizon enabled:

```
*A:ALA-1>config>service# info
----------------------------------------------
...
     vpls 800 customer 6001 vpn 700 create
          description "VPLS with split horizon for DSL"
          stp
               shutdown
          exit
          sap 1/1/3:100 split-horizon-group DSL-group1 create
               description "SAP for residential bridging"
          exit
          sap 1/1/3:200 split-horizon-group DSL-group1 create
               description "SAP for residential bridging"
          exit
          split-horizon-group DSL-group1
               description "Split horizon group for DSL"
          exit
          no shutdown
     exit
...
----------------------------------------------
*A:ALA-1>config>service#
```

Configuring MAC Learning Protection

To configure MAC learning protection, configure split horizon, MAC protection, and SAP parameters.

The following example displays a VPLS configuration with split horizon enabled:

```
A:ALA-48>config>service>vpls# info
----------------------------------------------
            description "IMA VPLS"
            split-horizon-group "DSL-group1" create
                restrict-protected-src
                restrict-unprotected-dst
            exit
            mac-protect
                 mac ff:ff:ff:ff:ff:ff
            exit
            sap 1/1/9:0 create
                ingress
                    scheduler-policy "SLA1"
                    qos 100 shared-queuing
                exit
                egress
                    scheduler-policy "SLA1"
                    filter ip 10
                exit
                restrict-protected-src
                arp-reply-agent
                host-connectivity-verify source-ip 143.144.145.1
            exit
...
----------------------------------------------
A:ALA-48>config>service>vpls#
```

Applying an Egress Multicast Group to a VPLS Service SAP

Use the following CLI syntax to apply an egress multicast group to a VPLS service SAP:

CLI Syntax: config>service>vpls *service-id* [customer *customer-id*] [vpn *vpn-id*] [mvpls]
sap *sap-id* [split-horizon-group *group-name*]
egress
multicast-group *group-name*

The following example displays a VPLS configuration with egress multicast group:

```
A:ALA-48>config>service>vpls# info
----------------------------------------------
            description "VPLS with split horizon for DSL"
            split-horizon-group "DSL-group1" create
                description "Split horizon group for DSL"
            exit
            stp
                shutdown
            exit
            sap 1/1/4:200 split-horizon-group "DSL-group1" create
                description "SAP for residential bridging"
            exit
            sap 1/1/3:100 split-horizon-group "DSL-group1" create
                description "SAP for residential bridging"
                egress
                   multicast-group "vpls-emg-1"
            exit
            no shutdown
----------------------------------------------
A:ALA-48>config>service>vpls#
```

Configuring SAP Subscriber Management Parameters

Use the following CLI syntax to configure subscriber management parameters on a VPLS service SAP. The policies and profiles that are referenced in the **def-sla-profile**, **def-sub-profile**, **non-sub-traffic**, and **sub-ident-policy** commands must already be configured in the **config>subscr-mgmt** context.

CLI Syntax:

```
config>service>vpls service-id
    sap sap-id [split-horizon-group group-name]
        sub-sla-mgmt
            def-sla-profile default-sla-profile-name
            def-sub-profile default-subscriber-profile-name
            mac-da-hashing
            multi-sub-sap [number-of-sub]
            no shutdown
            single-sub-parameters
                non-sub-traffic sub-profile sub-profile-name sla-
                    profile sla-profile-name [subscriber sub-ident-
                    string]
                profiled-traffic-only
            sub-ident-policy sub-ident-policy-name
```

The following example displays a subscriber management configuration:

```
A:ALA-48>config>service>vpls#
------------------------------------------
            description "Local VPLS"
            stp
                shutdown
            exit
            sap 1/2/2:0 create
                description "SAP for local service"
                sub-sla-mgmt
                    def-sla-profile "sla-profile1"
                    sub-ident-policy "SubIdent1"
                exit
            exit
            sap 1/1/5:0 create
                description "SAP for local service"
            exit
            no shutdown
------------------------------------------
A:ALA-48>config>service>vpls#
```

MSTP Control over Ethernet Tunnels

When MSTP is used to control VLANs, a range of VLAN IDs is normally used to specify the VLANs to be controlled.

If an Ethernet tunnel SAP is to be controlled by MSTP, the Ethernet Tunnel SAP ID needs to be within the VLAN range specified under the mst-instance.

```
vpls 400 customer 1 m-vpls create
            stp
                mode mstp
                mst-instance 111 create
                    vlan-range 1-100
                exit
                mst-name "abc"
                mst-revision 1
                no shutdown
            exit
            sap 1/1/1:0 create // untagged
            exit
            sap eth-tunnel-1 create
            exit
            no shutdown
        exit
        vpls 401 customer 1 create
            stp
                shutdown
            exit
            sap 1/1/1:12 create
            exit
            sap eth-tunnel-1:12 create
                // Ethernet tunnel SAP ID 12 falls within the VLAN
                // range for mst-instance 111
                eth-tunnel
                    path 1 tag 1000
                    path 8 tag 2000
                exit
            exit
            no shutdown
        exit
```

Configuring SDP Bindings

VPLS provides scaling and operational advantages. A hierarchical configuration eliminates the need for a full mesh of VCs between participating devices. Hierarchy is achieved by enhancing the base VPLS core mesh of VCs with access VCs (spoke) to form two tiers. Spoke SDPs are generally created between Layer 2 switches and placed at the Multi-Tenant Unit (MTU). The PE routers are placed at the service provider's Point of Presence (POP). Signaling and replication overhead on all devices is considerably reduced.

A spoke SDP is treated like the equivalent of a traditional bridge port where flooded traffic received on the spoke SDP is replicated on all other “ports” (other spoke and mesh SDPs or SAPs) and not transmitted on the port it was received (unless a split horizon group was defined on the spoke SDP, see section Configuring VPLS Spoke SDPs with Split Horizon on page 776).

A spoke SDP connects a VPLS service between two sites and, in its simplest form, could be a single tunnel LSP. A set of ingress and egress VC labels are exchanged for each VPLS service instance to be transported over this LSP. The PE routers at each end treat this as a virtual spoke connection for the VPLS service in the same way as the PE-MTU connections. This architecture minimizes the signaling overhead and avoids a full mesh of VCs and LSPs between the two metro networks.

A mesh SDP bound to a service is logically treated like a single bridge “port” for flooded traffic where flooded traffic received on any mesh SDP on the service is replicated to other “ports” (spoke SDPs and SAPs) and not transmitted on any mesh SDPs.

A VC-ID can be specified with the SDP-ID. The VC-ID is used instead of a label to identify a virtual circuit. The VC-ID is significant between peer SRs on the same hierarchical level. The value of a VC-ID is conceptually independent from the value of the label or any other datalink specific information of the VC.

Figure 36 displays an example of a distributed VPLS service configuration of spoke and mesh SDPs (uni-directional tunnels) between routers and MTUs.

Configuring Overrides on Service SAPs

The following output displays a service SAP queue override configuration example.

```
*A:ALA-48>config>service>vpls>sap# info
----------------------------------------------
...
                exit
                ingress
                    scheduler-policy "SLA1"
                    qos 100 multipoint-shared
                    queue-override
                        queue 1 create
                            rate 1500000 cir 2000
                        exit
                    exit
                exit
                egress
                    scheduler-policy "SLA1"
                    queue-override
                        queue 1 create
                            adaptation-rule pir max cir max
                        exit
                    exit
                    filter ip 10
                exit
----------------------------------------------
*A:ALA-48>config>service>vpls>sap#
```

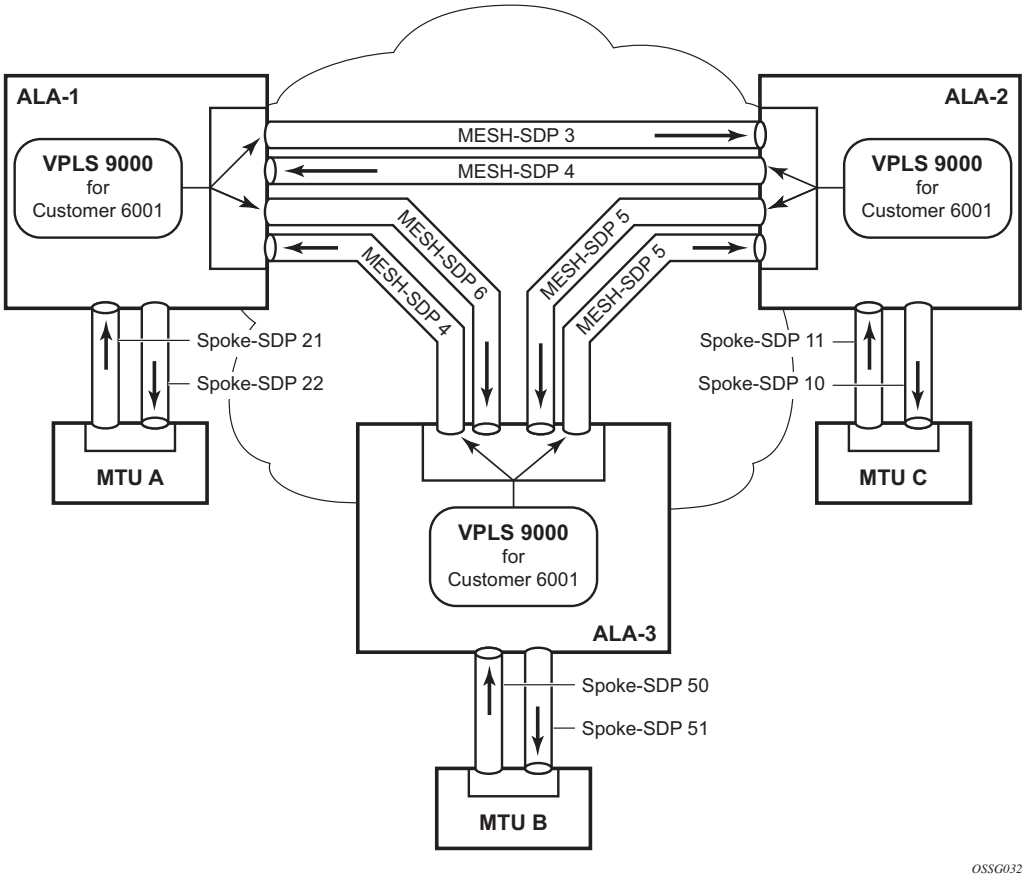

Figure 36: SDPs — Uni-Directional Tunnels

Use the following CLI syntax to create a mesh or spoke SDP bindings with a distributed VPLS service. SDPs must be configured prior to binding. Refer to Configuring an SDP on page 107 for information about creating SDPs.

Use the following CLI syntax to configure mesh SDP bindings:

CLI Syntax:
```
config>service# vpls service-id
 mesh-sdp sdp-id[:vc-id] [vc-type {ether | vlan}]
    egress
       filter {ip ip-filter-id|mac mac-filter-id}
       mfib-allowed-mda-destinations
             mda mda-id
       vc-label egress-vc-label
    ingress
       filter {ip ip-filter-id|mac mac-filter-id}
       vc-label ingress-vc-label
    no shutdown
    static-mac ieee-address
    vlan-vc-tag 0..4094
```

Use the following CLI syntax to configure spoke SDP bindings:

CLI Syntax:
```
config>service# vpls service-id
 spoke-sdp sdp-id:vc-id [vc-type {ether | vlan}] [split-hori-
    zon-group group-name]
    egress
       filter {ip ip-filter-id|mac mac-filter-id}
       vc-label egress-vc-label
    ingress
       filter {ip ip-filter-id|mac mac-filter-id}
       vc-label ingress-vc-label
    limit-mac-move[non-blockable]
    vlan-vc-tag 0..4094
    no shutdown
    static-mac ieee-address
    stp
       path-cost stp-path-cost
       priority stp-priority
       no shutdown
    vlan-vc-tag [0..4094]
```

The following displays SDP binding configurations for ALA-1, ALA-2, and ALA-3 for VPLS service ID 9000 for customer 6:

```
*A:ALA-1>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 create
            description "This is a distributed VPLS."
            def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/2/5:0 create
            exit
            spoke-sdp 2:22 create
            exit
            mesh-sdp 5:750 create
            exit
            mesh-sdp 7:750 create
            exit
            no shutdown
        exit
----------------------------------------------
*A:ALA-1>config>service#


*A:ALA-2>config>service# info
----------------------------------------------
...
        vpls 9000 customer 6 create
            description "This is a distributed VPLS."
```

```
            def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/1/2:22 create
            exit
            spoke-sdp 2:22 create
            exit
            mesh-sdp 5:750 create
            exit
            mesh-sdp 7:750 create
            exit
            no shutdown
        exit
-------------------------------------------


*A:ALA-3>config>service# info
-------------------------------------------
...
        vpls 9000 customer 6 create
            description "This is a distributed VPLS."
            def-mesh-vc-id 750
            stp
                shutdown
            exit
            sap 1/1/3:33 create
            exit
            spoke-sdp 2:22 create
            exit
            mesh-sdp 5:750 create
            exit
            mesh-sdp 7:750 create
            exit
            no shutdown
        exit
-------------------------------------------
*A:ALA-3>config>service#
```

Configuring Spoke SDP Specific STP Parameters

When a VPLS has STP enabled, each spoke SDP within the VPLS has STP enabled by default. The operation of STP on each spoke SDP is governed by:

Spoke SDP STP Administrative State

The administrative state of STP within a spoke SDP controls how BPDUs are transmitted and handled when received. The allowable states are:

- Spoke SDP Admin Up

  The default administrative state is *up* for STP on a spoke SDP. BPDUs are handled in the normal STP manner on a spoke SDP that is administratively up.

- Spoke SDP Admin Down

  An administratively down state allows a service provider to prevent a spoke SDP from becoming operationally blocked. BPDUs will not originate out the spoke SDP towards the customer.

  If STP is enabled on VPLS level, but disabled on the spoke SDP, received BPDUs are discarded. Discarding the incoming BPDUs allows STP to continue to operate normally within the VPLS service while ignoring the down spoke SDP. The specified spoke SDP will always be in an operationally forwarding state.

  NOTE: The administratively down state allows a loop to form within the VPLS.

CLI Syntax: `config>service>vpls>spoke-sdp>stp#`
`[no] shutdown`
Range: `shutdown` or `no shutdown`
Default: `no shutdown` (spoke SDP admin up)

Spoke SDP Virtual Port Number

The virtual port number uniquely identifies a spoke SDP within configuration BPDUs. The internal representation of a spoke SDP is unique to a system and has a reference space much bigger than the 12 bits definable in a configuration BPDU. STP takes the internal representation value of a spoke SDP and identifies it with it's own virtual port number that is unique to every other spoke SDP defined on the VPLS. The virtual port number is assigned at the time that the spoke SDP is added to the VPLS.
Since the order in which spoke SDPs are added to the VPLS is not preserved between reboots of the system, the virtual port number may change between restarts of the STP instance. To achieve consistency after a reboot, the virtual port number can be specified explicitly.

CLI Syntax: `config>service>vpls>spoke-sdp# stp`
`port-num` *`number`*
Range: 1 — 2047
Default: (automatically generated)
Restore Default: `no port-num`

Spoke SDP Priority

Spoke SDP priority allows a configurable tie breaking parameter to be associated with a spoke SDP. When configuration BPDUs are being received, the configured spoke SDP priority will be used in some circumstances to determine whether a spoke SDP will be designated or blocked.

In traditional STP implementations (802.1D-1998), this field is called the port priority and has a value of 0 to 255. This field is coupled with the port number (0 to 255 also) to create a 16 bit value. In the latest STP standard (802.1D-2004) only the upper 4 bits of the port priority field are used to encode the spoke SDP priority. The remaining 4 bits are used to extend the port ID field into a 12 bit virtual port number field. The virtual port number uniquely references a spoke SDP within the STP instance. See Spoke SDP Virtual Port Number on page 770 for details on the virtual port number.

STP computes the actual spoke SDP priority by taking the configured priority value and masking out the lower four bits. The result is the value that is stored in the spoke SDP priority parameter. For instance, if a value of 0 was entered, masking out the lower 4 bits would result in a parameter value of 0. If a value of 255 was entered, the result would be 240.

The default value for spoke SDP priority is 128. This parameter can be modified within a range of 0 to 255, 0 being the highest priority. Masking causes the values actually stored and displayed to be 0 to 240, in increments of 16.

CLI Syntax: `config>service>vpls>spoke-sdp>stp#`
`priority` *`stp-priority`*
Range: 0 to 255 (240 largest value, in increments of 16)
Default: 128
Restore Default: `no priority`

Spoke SDP Path Cost

The spoke SDP path cost is used by STP to calculate the path cost to the root bridge. The path cost in BPDUs received on the root port is incremented with the configured path cost for that spoke SDP. When BPDUs are sent out other egress spoke SDPs, the newly calculated root path cost is used.

STP suggests that the path cost is defined as a function of the link bandwidth. Since spoke SDPs are controlled by complex queuing dynamics, the STP path cost is a purely static configuration.

The default value for spoke SDP path cost is 10. This parameter can be modified within a range of 1 to 200000000 (1 is the lowest cost).

CLI Syntax: `config>service>vpls>spoke-sdp>stp#`
`path-cost` *`stp-path-cost`*
Range: 1 to 200000000
Default: 10
Restore Default: `no path-cost`

Spoke SDP Edge Port

The spoke SDP `edge-port` command is used to reduce the time it takes a spoke SDP to reach the forwarding state when the spoke SDP is on the edge of the network, and thus has no further STP bridge to handshake with.

The `edge-port` command is used to initialize the internal OPER_EDGE variable. At any time, when OPER_EDGE is false on a spoke SDP, the normal mechanisms are used to transition to the forwarding state (see Forward Delay on page 746). When OPER_EDGE is true, STP assumes that the remote end agrees to transition to the forwarding state without actually receiving a BPDU with an agreement flag set.

The OPER_EDGE variable will dynamically be set to false if the spoke SDP receives BPDUs (the configured edge-port value does not change). The OPER_EDGE variable will dynamically be set to true if auto-edge is enabled and STP concludes there is no bridge behind the spoke SDP.

When STP on the spoke SDP is administratively disabled and re-enabled, the OPER_EDGE is re-initialized to the spoke SDP configured for edge-port.

Valid values for spoke SDP edge-port are enabled and disabled with disabled being the default.

CLI Syntax: `config>service>vpls>spoke-sdp>stp#`
`[no] edge-port`
Default: `no edge-port`

Spoke SDP Auto Edge

The spoke SDP `edge-port` command is used to instruct STP to dynamically decide whether the spoke SDP is connected to another bridge.

If auto-edge is enabled, and STP concludes there is no bridge behind the spoke SDP, the OPER_EDGE variable will dynamically be set to true. If auto-edge is enabled, and a BPDU is received, the OPER_EDGE variable will dynamically be set to true (see Spoke SDP Edge Port on page 771).

Valid values for spoke SDP auto-edge are enabled and disabled with enabled being the default.

CLI Syntax: `config>service>vpls>spoke-sdp>stp#`
`[no] auto-edge`
Default: `auto-edge`

Spoke SDP Link Type

The spoke SDP `link-type` command instructs STP on the maximum number of bridges behind this spoke SDP. If there is only a single bridge, transitioning to forwarding state will be based on handshaking (fast transitions). If more than two bridges are connected by a shared media, their spoke SDPs should all be configured as shared, and timer-based transitions are used.

Valid values for spoke SDP link-type are shared and pt-pt with pt-pt being the default.

CLI Syntax: `config>service>vpls>spoke-sdp>stp#`
`link-type {pt-pt|shared}`
Default: `link-type pt-pt`
Restore Default: `no link-type`

Spoke SDP STP Operational States

The operational state of STP within a spoke SDP controls how BPDUs are transmitted and handled when received. Defined states are:

- Operationally Disabled on page 773
- Operationally Discarding on page 773
- Operationally Learning on page 773
- Operationally Forwarding on page 774

Operationally Disabled

Operationally disabled is the normal operational state for STP on a spoke SDP in a VPLS that has any of the following conditions:

- VPLS state administratively down
- Spoke SDP state administratively down
- Spoke SDP state operationally down

If the spoke SDP enters the operationally up state with the STP administratively up and the spoke SDP STP state is up, the spoke SDP will transition to the STP spoke SDP discarding state.

When, during normal operation, the router detects a downstream loop behind a spoke SDP, BPDUs can be received at a very high rate. To recover from this situation, STP will transition the spoke SDP to a disabled state for the configured forward-delay duration.

Operationally Discarding

A spoke SDP in the discarding state only receives and sends BPDUs, building the local proper STP state for each spoke SDP while not forwarding actual user traffic. The duration of the discarding state is explained in section Forward Delay on page 746.
Note: in previous versions of the STP standard, the discarding state was called a blocked state.

Operationally Learning

The learning state allows population of the MAC forwarding table before entering the forwarding state. In this state no user traffic is forwarded.

Operationally Forwarding

Configuration BPDUs are sent out a spoke SDP in the forwarding state. Layer 2 frames received on the spoke SDP are source learned and destination forwarded according to the FIB. Layer 2 frames received on other forwarding interfaces and destined for the spoke SDP are also forwarded.

Spoke SDP BPDU Encapsulation States

IEEE 802.1D (referred as dot1d) and Cisco's per VLAN Spanning Tree (PVST) BPDU encapsulations are supported on a per spoke SDP basis. STP is associated with a VPLS service like PVST is per VLAN. The main difference resides in the Ethernet and LLC framing and a type-length-value (TLV) field trailing the BPDU.

Table 8 shows differences between dot1D and PVST Ethernet BPDU encapsulations based on the interface encap-type field:

Table 8: Spoke SDP BPDU Encapsulation States

Field	dot1d encap-type null	dot1d encap-type dot1q	PVST encap-type null	PVST encap-type dot1q
Destination MAC	01:80:c2:00:00:00	01:80:c2:00:00:00	N/A	01:00:0c:cc:cc:cd
Source MAC	Sending Port MAC	Sending Port MAC	N/A	Sending Port MAC
EtherType	N/A	0x81 00	N/A	0x81 00
Dot1p and CFI	N/A	0xe	N/A	0xe
Dot1q	N/A	VPLS spoke SDP ID	N/A	VPLS spoke SDP encap value
Length	LLC Length	LLC Length	N/A	LLC Length
LLC DSAP SSAP	0x4242	0x4242	N/A	0xaaaa (SNAP)
LLC CNTL	0x03	0x03	N/A	0x03
SNAP OUI	N/A	N/A	N/A	00 00 0c (Cisco OUI)
SNAP PID	N/A	N/A	N/A	01 0b
CONFIG or TCN BPDU	Standard 802.1d	Standard 802.1d	N/A	Standard 802.1d
TLV: Type & Len	N/A	N/A	N/A	58 00 00 00 02
TLV: VLAN	N/A	N/A	N/A	VPLS spoke SDP encap value
Padding	As Required	As Required	N/A	As Required

Each spoke SDP has a Read Only operational state that shows which BPDU encapsulation is currently active on the spoke SDP. The following states apply:

- **Dot1d** specifies that the switch is currently sending IEEE 802.1D standard BPDUs. The BPDUs will be tagged or non-tagged based on the encapsulation type of the egress interface and the encapsulation value defined in the spoke SDP. A spoke SDP defined on an interface with encapsulation type dot1q will continue in the dot1d BPDU encapsulation state until a PVST encapsulated BPDU is received, after which the spoke SDP will convert to the PVST encapsulation state. Each received BPDU must be properly IEEE 802.1q tagged if the interface encapsulation type is defined to dot1q.
- **PVST** specifies that the switch is currently sending proprietary encapsulated BPDUs. PVST BPDUs are only supported on Ethernet interfaces with the encapsulation type set to dot1q. The spoke SDP continues in the PVST BPDU encapsulation state until a dot1d encapsulated BPDU is received, in which case the spoke SDP reverts to the dot1d encapsulation state. Each received BPDU must be properly IEEE 802.1q tagged with the encapsulation value defined for the spoke SDP.

Dot1d is the initial and only spoke SDP BPDU encapsulation state for spoke SDPs defined on Ethernet interface with encapsulation type set to null.

Each transition between encapsulation types optionally generates an alarm that can be logged and optionally transmitted as an SNMP trap.

Configuring VPLS Spoke SDPs with Split Horizon

To configure spoke SDPs with a split horizon group, add the split-horizon-group parameter when creating the spoke SDP. Traffic arriving on a SAP or spoke SDP within a split horizon group will not be copied to other SAPs or spoke SDPs in the same split horizon group.

The following example displays a VPLS configuration with split horizon enabled:

```
*A:ALA-1>config>service# info
----------------------------------------------
...
vpls 800 customer 6001 vpn 700 create
     description "VPLS with split horizon for DSL"
     stp
          shutdown
     exit
     spoke-sdp 51:15 split-horizon-group DSL-group1 create
     exit
     split-horizon-group DSL-group1
          description "Split horizon group for DSL"
     exit
     no shutdown
exit
...
----------------------------------------------
*A:ALA-1>config>service#
```

Configuring VPLS Redundancy

This section discusses the following service management tasks:

Creating a Management VPLS for SAP Protection

This section provides a brief overview of the tasks that must be performed to configure a management VPLS for SAP protection and provides the CLI commands, see Figure 37. The tasks below should be performed on both nodes providing the protected VPLS service.

Before configuring a management VPLS, first read VPLS Redundancy on page 643 for an introduction to the concept of management VPLS and SAP redundancy.

1. Create an SDP to the peer node.
2. Create a management VPLS.
3. Define a SAP in the m-vpls on the port towards the MTU. Note that the port must be dot1q or qinq tagged. The SAP corresponds to the (stacked) VLAN on the MTU in which STP is active.
4. Optionally modify STP parameters for load balancing .
5. Create a mesh SDP in the m-vpls using the SDP defined in Step 1. Ensure that this mesh SDP runs over a protected LSP (see note below).
6. Enable the management VPLS service and verify that it is operationally up.
7. Create a list of VLANs on the port that are to be managed by this management VPLS.
8. Create one or more user VPLS services with SAPs on VLANs in the range defined by Step 6.

Note: The mesh SDP should be protected by a backup LSP or Fast Reroute. If the mesh SDP were to go down, STP on both nodes would go to “forwarding” state and a loop would occur.

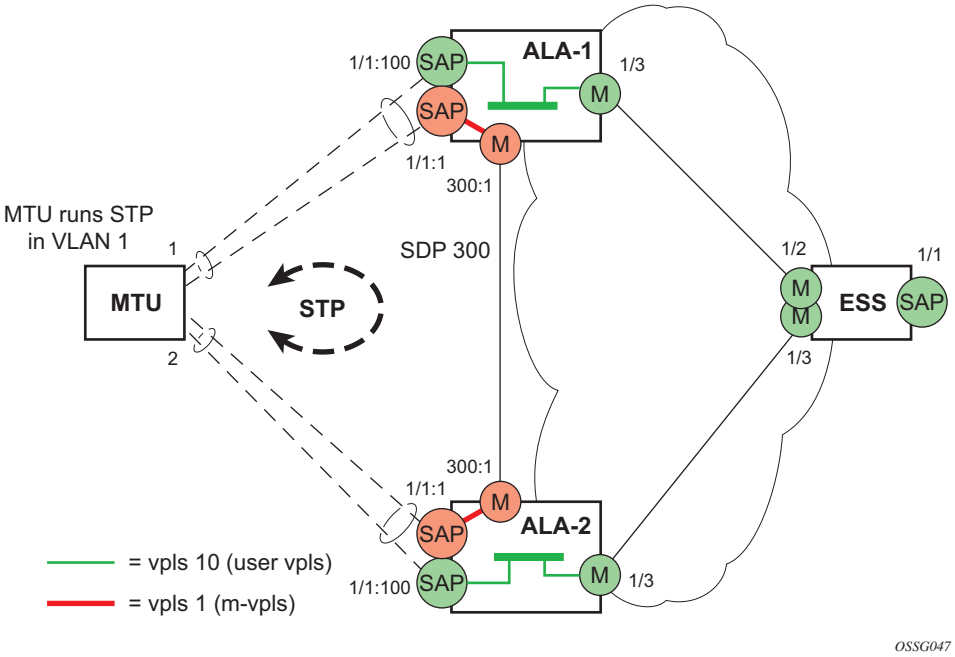

Figure 37: Example Configuration for Protected VPLS SAP

Use the following CLI syntax to create a management VPLS:

CLI Syntax:
```
config>service# sdp sdp-id mpls create
    far-end ip-address
    lsp lsp-name
    no shutdown
```

CLI Syntax:
```
vpls service-id customer customer-id [m-vpls] create
    description description-string
    sap sap-id create
       managed-vlan-list
          range vlan-range
    mesh-sdp sdp-id:vc-id create
    stp
    no shutdown
```

The following example displays a VPLS configuration:

```
*A:ALA-1>config>service# info
----------------------------------------------
...
       sdp 300 mpls create
            far-end 10.0.0.20
            lsp "toALA-A2"
            no shutdown
       exit
       vpls 1 customer 1 m-vpls create
            sap 1/1/1:1 create
```

```
                managed-vlan-list
                    range 100-1000
                exit
            exit
            mesh-sdp 300:1 create
            exit
            stp
            exit
            no shutdown
        exit
...
-----------------------------------------------
*A:ALA-1>config>service#
```

Creating a Management VPLS for Spoke SDP Protection

This section provides a brief overview of the tasks that must be performed to configure a management VPLS for spoke SDP protection and provides the CLI commands, see Figure 38. The tasks below should be performed on all four nodes providing the protected VPLS service. Before configuring a management VPLS, first read Configuring a VPLS SAP on page 750 for an introduction to the concept of management VPLS and spoke SDP redundancy.

1. Create an SDP to the local peer node (node ALA-A2 in the example below).
2. Create an SDP to the remote peer node (node ALA-B1 in the example below).
3. Create a management VPLS.
4. Create a spoke SDP in the m-vpls using the SDP defined in Step 1. Ensure that this mesh SDP runs over a protected LSP (see note below).
5. Enable the management VPLS service and verify that it is operationally up.
6. Create a spoke SDP in the m-vpls using the SDP defined in Step 2.
   Optionally, modify STP parameters for load balancing (see Configuring Load Balancing with Management VPLS on page 783).
7. Create one or more user VPLS services with spoke SDPs on the tunnel SDP defined by Step 2.

As long as the user spoke SDPs created in step 7are in this same tunnel SDP with the management spoke SDP created in step 6, the management VPLS will protect them.

The SDP should be protected by, for example, a backup LSP or Fast Reroute. If the SDP were to go down, STP on both nodes would go to “forwarding” state and a loop would occur.

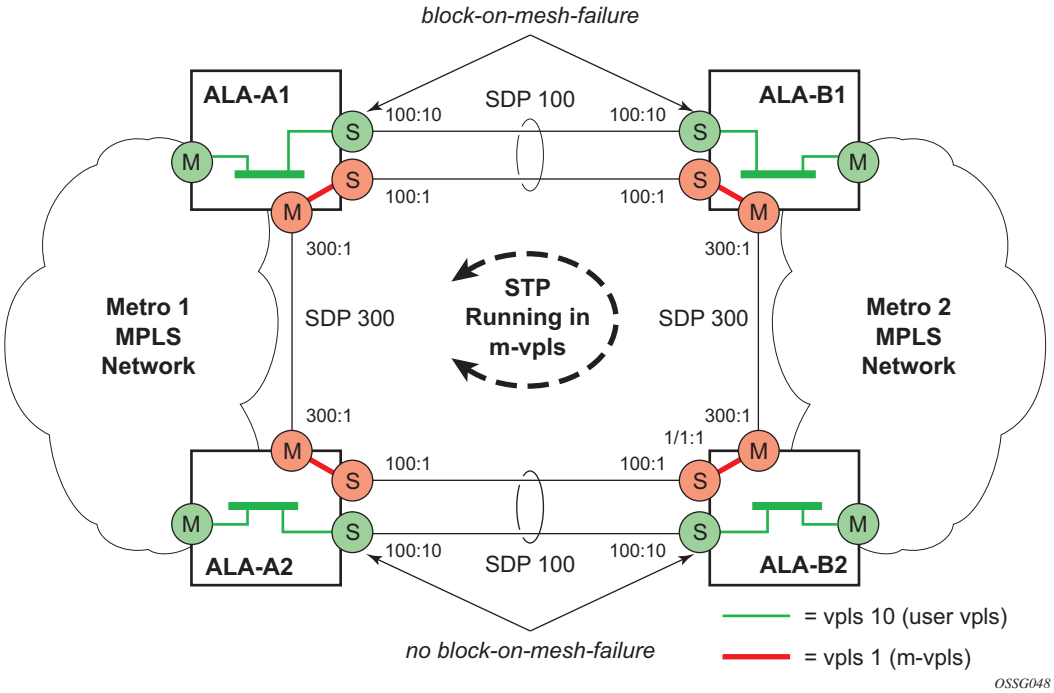

Figure 38: Example Configuration for Protected VPLS Spoke SDP

Use the following CLI syntax to create a management VPLS for spoke SDP protection:

CLI Syntax:
```
config>service# sdp sdp-id mpls create
 far-end ip-address
 lsp lsp-name
 no shutdown
```

CLI Syntax:
```
vpls service-id customer customer-id [m-vpls] create
 description description-string
 mesh-sdp sdp-id:vc-id create
 spoke-sdp sdp-id:vc-id create
 stp
 no shutdown
```

The following example displays a VPLS configuration:

```
*A:ALA-A1>config>service# info
----------------------------------------------
...
       sdp 100 mpls create
            far-end 10.0.0.30
            lsp "toALA-B1"
            no shutdown
       exit
       sdp 300 mpls create
            far-end 10.0.0.20
            lsp "toALA-A2"
            no shutdown
       exit
       vpls 101 customer 1 m-vpls create
            spoke-sdp 100:1 create
            exit
            meshspoke-sdp 300:1 create
            exit
            stp
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-A1>config>service#
```

Configuring Load Balancing with Management VPLS

With the concept of management VPLS, it is possible to load balance the user VPLS services across the two protecting nodes. This is done by creating two management VPLS instances, where both instances have different active QinQ spokes (by changing the STP path-cost). When different user VPLS services are associated with either the two management VPLS services, the traffic will be split across the two QinQ spokes. Load balancing can be achieved in both the SAP protection and spoke SDP protection scenarios.

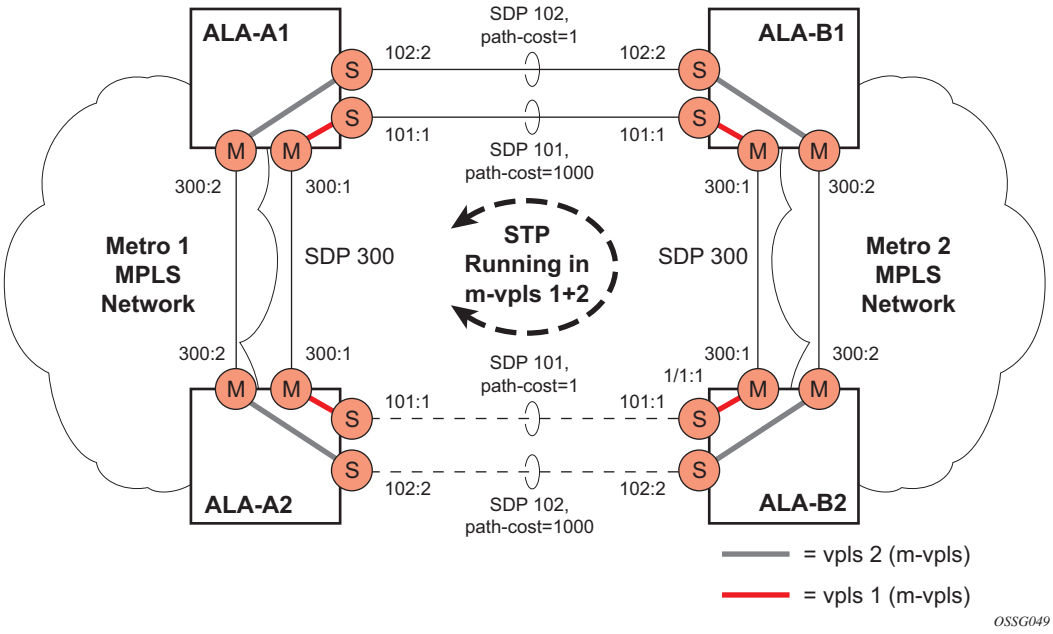

Figure 39: Example Configuration for Load Balancing Across Two Protected VPLS Spoke SDPs

Use the following CLI syntax to create a load balancing across two management VPLS instances:

CLI Syntax:
```
config>service# sdp sdp-id mpls create
 far-end ip-address
 lsp lsp-name
 no shutdown
```

CLI Syntax:
```
vpls service-id customer customer-id [m-vpls] create
 description description-string
 mesh-sdp sdp-id:vc-id create
 spoke-sdp sdp-id:vc-id create
    stp
       path-cost
 stp
 no shutdown
```

Note: the STP path costs in each peer node should be reversed.

The following example displays the VPLS configuration on ALA-A1 (top left, IP address 10.0.0.10):

```
*A:ALA-A1>config>service# info
----------------------------------------------
...
      sdp 101 mpls create
            far-end 10.0.0.30
            lsp "1toALA-B1"
            no shutdown
      exit
      sdp 102 mpls create
            far-end 10.0.0.30
            lsp "2toALA-B1"
            no shutdown
      exit
 ...
     vpls 101 customer 1 m-vpls create
            spoke-sdp 101:1 create
               stp
                  path-cost 1
               exit
            exit
            mesh-sdp 300:1 create
            exit
            stp
            exit
           no shutdown
      exit
      vpls 102 customer 1 m-vpls create
            spoke-sdp 102:2 create
               stp
                  path-cost 1000
               exit
            exit
            mesh-sdp 300:2 create
            exit
            stp
            exit
            no shutdown
      exit
...
----------------------------------------------
*A:ALA-A1>config>service#
```

The following example displays the VPLS configuration on ALA-A2 (bottom left, IP address 10.0.0.20):

```
*A:ALA-A2>config>service# info
----------------------------------------------
...
      sdp 101 mpls create
            far-end 10.0.0.40
            lsp "1toALA-B2"
            no shutdown
      exit
      sdp 102 mpls create
            far-end 10.0.0.40
            lsp "2toALA-B2"
            no shutdown
      exit
 ...
     vpls 101 customer 1 m-vpls create
            spoke-sdp 101:1 create
               stp
                  path-cost 1000
               exit
            exit
            mesh-sdp 300:1 create
            exit
            stp
            exit
           no shutdown
      exit
      vpls 102 customer 1 m-vpls create
            spoke-sdp 102:2 create
               stp
                  path-cost 1
               exit
            exit
            mesh-sdp 300:2 create
            exit
            stp
            exit
            no shutdown
      exit
...
----------------------------------------------
*A:ALA-A2>config>service#
```

The following example displays the VPLS configuration on ALA-A3 (top right, IP address 10.0.0.30):

```
*A:ALA-A1>config>service# info
----------------------------------------------
...
      sdp 101 mpls create
            far-end 10.0.0.10
            lsp "1toALA-A1"
            no shutdown
      exit
      sdp 102 mpls create
            far-end 10.0.0.10
            lsp "2toALA-A1"
            no shutdown
      exit
 ...
     vpls 101 customer 1 m-vpls create
            spoke-sdp 101:1 create
               stp
                  path-cost 1
               exit
            exit
            mesh-sdp 300:1 create
            exit
            stp
            exit
            no shutdown
      exit
      vpls 102 customer 1 m-vpls create
            spoke-sdp 102:2 create
               stp
                  path-cost 1000
               exit
            exit
            mesh-sdp 300:2 create
            exit
            stp
            exit
            no shutdown
      exit
...
----------------------------------------------
*A:ALA-A1>config>service#
```

The following example displays the VPLS configuration on ALA-A4 (bottom right, IP address 10.0.0.40):

```
*A:ALA-A2>config>service# info
----------------------------------------------
...
      sdp 101 mpls create
            far-end 10.0.0.20
            lsp "1toALA-B2"
            no shutdown
      exit
      sdp 102 mpls create
            far-end 10.0.0.20
            lsp "2toALA-B2"
            no shutdown
      exit
 ...
     vpls 101 customer 1 m-vpls create
            spoke-sdp 101:1 create
               stp
                  path-cost 1000
               exit
            exit
            mesh-sdp 300:1 create
            exit
            stp
            exit
            no shutdown
      exit
      vpls 102 customer 1 m-vpls create
            spoke-sdp 102:2 create
               stp
                  path-cost 1
               exit
            exit
            mesh-sdp 300:2 create
            exit
            stp
            exit
            no shutdown
      exit
...
----------------------------------------------
*A:ALA-A2>config>service#
```

Configuring Selective MAC Flush

Use the following CLI syntax to enable selective MAC Flush in a VPLS.

CLI Syntax: `config>service# vpls` *`service-id`*
`send-flush-on-failure`

Use the following CLI syntax to disable selective MAC Flush in a VPLS.

CLI Syntax: `config>service# vpls` *`service-id`*
`no send-flush-on-failure`

Configuring Multi-Chassis Endpoints

The following output displays configuration examples of multi-chassis redundancy and the VPLS configuration. The configurations in the graphics depicted in Inter-Domain VPLS Resiliency Using Multi-Chassis Endpoints on page 646 are expressed in this output.

Node Mappig to figures the document:

- PE3 = Dut-B
- PE3' = Dut-C
- PE1 = Dut-D
- PE2 = Dut-E

PE3

```
*A:Dut-B>config>redundancy>multi-chassis# info
----------------------------------------------
            peer 3.1.1.3 create
                peer-name "Dut-C"
                description "mcep-basic-tests"
                source-address 2.1.1.2
                mc-endpoint
                    no shutdown
                    bfd-enable
                    system-priority 50
                exit
                no shutdown
            exit
----------------------------------------------
*A:Dut-B>config>redundancy>multi-chassis#


*A:Dut-B>config>service>vpls# info
----------------------------------------------
            fdb-table-size 20000
            send-flush-on-failure
            stp
                shutdown
            exit
            endpoint "mcep-t1" create
                no suppress-standby-signaling
                block-on-mesh-failure
                mc-endpoint 1
                    mc-ep-peer Dut-C
                exit
            exit
            mesh-sdp 201:1 vc-type vlan create
            exit
            mesh-sdp 211:1 vc-type vlan create
            exit
            spoke-sdp 221:1 vc-type vlan endpoint "mcep-t1" create
                stp
```

```
                    shutdown
                exit
                block-on-mesh-failure
                precedence 1
            exit
            spoke-sdp 231:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
                precedence 2
            exit
            no shutdown
-----------------------------------------------
*A:Dut-B>config>service>vpls#
```

PE3' Dut-C

```
:Dut-C>config>redundancy>multi-chassis# info
-----------------------------------------------
            peer 2.1.1.2 create
                peer-name "Dut-B"
                description "mcep-basic-tests"
                source-address 3.1.1.3
                mc-endpoint
                    no shutdown
                    bfd-enable
                    system-priority 21
                exit
                no shutdown
            exit
-----------------------------------------------
*A:Dut-C>config>redundancy>multi-chassis#

*A:Dut-C>config>service>vpls# info
-----------------------------------------------
            fdb-table-size 20000
            send-flush-on-failure
            stp
                shutdown
            exit
            endpoint "mcep-t1" create
                no suppress-standby-signaling
                block-on-mesh-failure
                mc-endpoint 1
                    mc-ep-peer Dut-B
                exit
            exit
            mesh-sdp 301:1 vc-type vlan create
            exit
            mesh-sdp 311:1 vc-type vlan create
            exit
            spoke-sdp 321:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
                precedence 3
```

```
            exit
            spoke-sdp 331:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
            exit
            no shutdown
-----------------------------------------------
*A:Dut-C>config>service>vpls#
```

PE1 Dut-D

```
*A:Dut-D>config>redundancy>multi-chassis# info
-----------------------------------------------
            peer 5.1.1.5 create
                peer-name "Dut-E"
                description "mcep-basic-tests"
                source-address 4.1.1.4
                mc-endpoint
                    no shutdown
                    bfd-enable
                    system-priority 50
                    passive-mode
                exit
                no shutdown
            exit
-----------------------------------------------
*A:Dut-D>config>redundancy>multi-chassis#

*A:Dut-D>config>service>vpls# info
-----------------------------------------------
            fdb-table-size 20000
            propagate-mac-flush
            stp
                shutdown
            exit
            endpoint "mcep-t1" create
                block-on-mesh-failure
                mc-endpoint 1
                    mc-ep-peer Dut-E
                exit
            exit
            mesh-sdp 401:1 vc-type vlan create
            exit
            spoke-sdp 411:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
                precedence 2
            exit
            spoke-sdp 421:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
                precedence 1
```

```
            exit
            mesh-sdp 431:1 vc-type vlan create
            exit
            no shutdown
-----------------------------------------------
*A:Dut-D>config>service>vpls#
```

PE2 Dut-E

```
*A:Dut-E>config>redundancy>multi-chassis# info
-----------------------------------------------
            peer 4.1.1.4 create
                peer-name "Dut-D"
                description "mcep-basic-tests"
                source-address 5.1.1.5
                mc-endpoint
                    no shutdown
                    bfd-enable
                    system-priority 22
                    passive-mode
                exit
                no shutdown
            exit
-----------------------------------------------
*A:Dut-E>config>redundancy>multi-chassis#

*A:Dut-E>config>service>vpls# info
-----------------------------------------------
            fdb-table-size 20000
            propagate-mac-flush
            stp
                shutdown
            exit
            endpoint "mcep-t1" create
                block-on-mesh-failure
                mc-endpoint 1
                    mc-ep-peer Dut-D
                exit
            exit
            spoke-sdp 501:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
                precedence 3
            exit
            spoke-sdp 511:1 vc-type vlan endpoint "mcep-t1" create
                stp
                    shutdown
                exit
                block-on-mesh-failure
            exit
            mesh-sdp 521:1 vc-type vlan create
            exit
            mesh-sdp 531:1 vc-type vlan create
            exit
            no shutdown
-----------------------------------------------
*A:Dut-E>config>service>vpls#
```

ATM/Frame Relay PVC Access and Termination on a VPLS Service

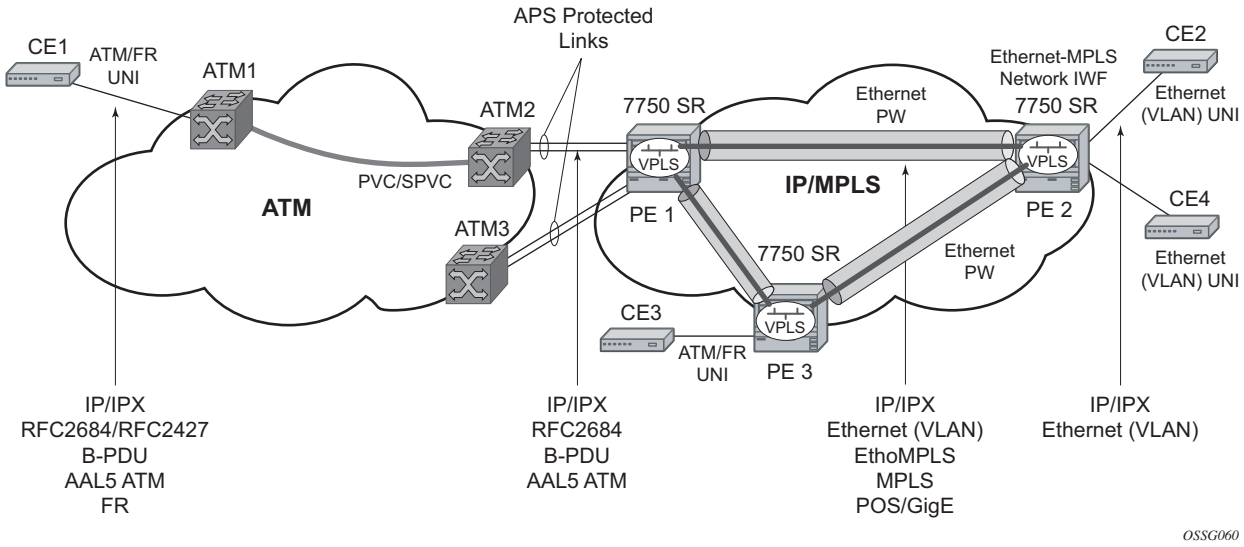

Figure 40: ATM/Frame Relay PVC Access and Termination on a VPLS Example

The application is depicted in above provides access to a VPLS service to Frame Relay and ATM users connected either directly or through an ATM access network to a 7750 PE node. The 7750 SR supports a Frame Relay or an ATM VC-delimited Service Access Point (SAP) terminating on a VPLS service.

RFC 2427-encapsulated or RFC 2684-encapsulated untagged Ethernet/802.3 frames (with or without Frame Check Sequence (FCS)) or BPDUs from a customer's bridge device are received on a given SAP over an ATM or Frame Relay interface on the 7750 SR. The Frame Relay or ATM-related encapsulation is stripped and the frames (without FCS) are forwarded towards destination SAPs either locally, or using SDPs associated with the VPLS service (as dictated by destination MAC address VPLS processing). In the egress direction, the received untagged frames are encapsulated into RFC 2427 or RFC 2684 (no Q-tags are added, no FCS in the forwarded frame) and sent over ATM or a FR VC towards the customer CPE.

When AAL5 RFC2427/2684 encapsulated tagged frames are received from the customer's bridge on an FR/ATM SAP, the tags are transparent and the frames are processed as described above with the exception that the frames forwarded towards the destination(s) will have the received tags preserved. Similarly in the egress direction, the received tagged Ethernet frames are encapsulated as is (i.e. Q-tags are again transparent and preserved) into RFC 2427/2684 and sent over the FR/ATM PVC towards the customer CPE. Note that since the tagging is transparent, the 7750 SR performs unqualified MAC learning (for example, MAC addresses are learned without reference to VLANs they are associated with). Because of that, MAC addresses used must be unique across all the VLANs used by the customer for a given VPLS service instance. If a customer wants a per-

VLAN separation, then the VLAN traffic that needs to be separated must come on different VCs (different SAPs) associated with different VPLS service instances.

All VPLS functionality available on the 7750 SR is applicable to FR and ATM-delimited VPLS SAPs. For example, bridged PDUs received over ATM SAP can be tunneled through or dropped; all Forwarding Information Base functionality applies; packet level QoS and MAC filtering applies; etc. Also, split horizon groups are applicable to ATM SAPs terminating on VPLS. In other words, frame forwarding between ATM SAPs, also referred to as VCI-to-VCI forwarding, within the same group is disabled.

The Ethernet pseudowire is established using Targeted LDP (TLDP) signaling and uses the ether, vlan, or vpls VC type on the SDP. The SDP can be an MPLS or a GRE type.

Configuring BGP Auto-Discovery

This section provides important information to explain the different configuration options used to populate the required BGP AD and generate the LDP generalized pseudowire-ID FEC fields. There are a large number of configuration options that are available with the this feature. Not all these configurations option are required to start using BGP AD. At the end of this section, it will be apparent that a very simple configuration will automatically generate the required values used by BGP and LDP. In most cases, deployments will provide full mesh connectivity between all nodes across a VPLS instance. However, capabilities are available to influence the topology and build hierarchies or hub and spoke models.

Configuration Steps

Using Figure 41, assume PE6 was previously configured with VPLS 100 as indicated by the configurations lines in the upper right. The BGP AD process will commence after PE134 is configured with the VPLS 100 instance as shown in the upper left. This shows a very basic and simple BGP AD configuration. The minimum requirement for enabling BGP AD on a VPLS instance is configuring the VPLS-ID and point to a pseudowire template.

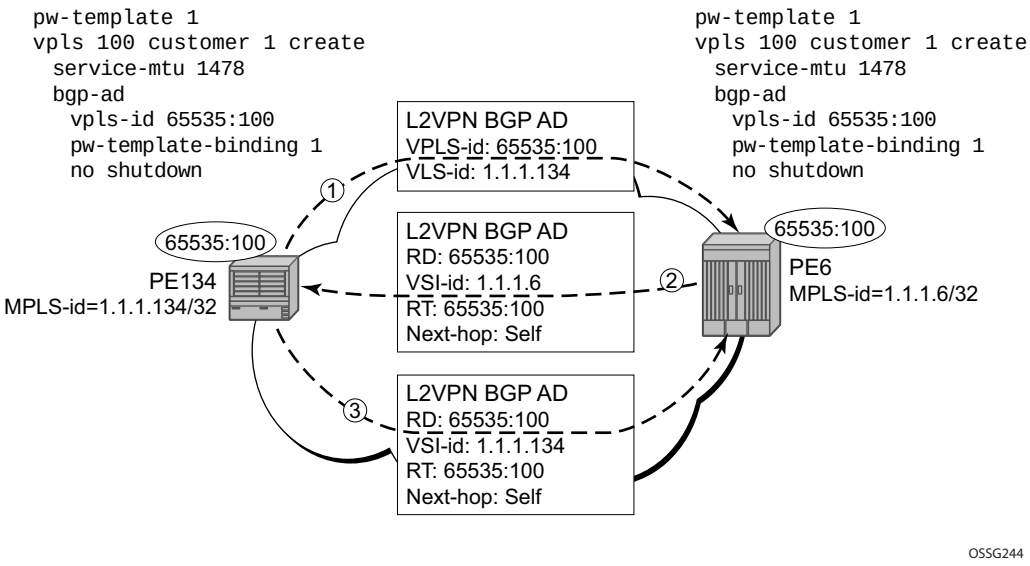

Figure 41: BGP AD Configuration Example

In many cases, VPLS connectivity is based on a pseudowire mesh. To reduce the configuration requirement, the BGP values can be automatically generated using the VPLS-ID and the MPLS router-ID. By default, the lower six bytes of the VPLS-ID are used to generate the RD and the RT

values. The VSI-ID value is generated from the MPLS router-ID. All of these parameters are configurable and can be coded to suit requirements and build different topologies.

```
PE134>config>service>vpls>bgp-ad#
 [no] pw-template-bi* - Configure pw-template bind policy
 [no] route-target    - Configure route target
 [no] shutdown        - Administratively enable/disable BGP auto-discovery
      vpls-id         - Configure VPLS-ID
 [no] vsi-export      - VSI export route policies
      vsi-id          + Configure VSI-id
 [no] vsi-import      - VSI import route policies
```

Figure 42: BGP-AD CLI Command Tree

A helpful command displays the service information, the BGP parameters and the SDP bindings in use. When the discovery process is completed successfully each endpoint will have an entry for the service.

```
PE134>#  show service l2-route-table
===============================================================================
Services: L2 Route Information - Summary Service
===============================================================================
Svc Id     L2-Routes (RD-Prefix)                Next Hop        Origin
               Sdp Bind Id
-------------------------------------------------------------------------------
100        65535:100-1.1.1.6                    1.1.1.6         BGP-L2
               17406:4294967295
-------------------------------------------------------------------------------
No. of L2 Route Entries: 1
===============================================================================
PERs6>#

PERs6>#  show service l2-route-table
===============================================================================
Services: L2 Route Information - Summary Service
===============================================================================
Svc Id     L2-Routes (RD-Prefix)                Next Hop        Origin
               Sdp Bind Id
-------------------------------------------------------------------------------
100        65535:100-1.1.1.134                  1.1.1.134       BGP-L2
               17406:4294967295
-------------------------------------------------------------------------------
No. of L2 Route Entries: 1
===============================================================================
PERs6>#
```

When only one of the endpoints has an entry for the service in the l2-routing-table, it is most likely a problem with the RT values used for import and export. This would most likely happen when different import and export RT values are configured using a router policy or the route-target command.

Service specific commands continue to be available to display service specific information, including status.

```
PERs6# show service sdp-using
===============================================================================
SDP Using
===============================================================================
SvcId       SdpId               Type    Far End        Opr S* I.Label  E.Label
-------------------------------------------------------------------------------
100         17406:4294967295    BgpAd   1.1.1.134      Up     131063   131067
-------------------------------------------------------------------------------
Number of SDPs : 1
===============================================================================
* indicates that the corresponding row element may have been truncated.
```

BGP AD will advertise the VPLS-ID in the extended community attribute, VSI-ID in the NLRI and the local PE id in the BGP next hop. At the receiving PE, the VPLS-ID is compared against locally provisioned information to determine whether the two PEs share a common VPLS. If it is found that they do, the BGP information is used in the signaling phase (see Configuring BGP VPLS on page 802).

LDP Signaling

T-LDP is triggered once the VPN endpoints have been discovered using BGP. The T-LDP session between the PEs is established when one does not exist. The far-end IP address required for the T-LDP identification is gleaned from the BGP AD next hop information. The pw-template and pw-template-binding configuration statements are used to establish the automatic SDP or to map to the appropriate SDP. The FEC129 content is built using the following values:

- AGI from the locally configured VPLS-ID.
- The SAII from the locally configured VSI-ID.
- The TAII from the VSI-ID contained in the last 4 bytes of the received BGP NLRI.

Figure 43 below shows the different detailed phases of the LDP signaling path, post BGP AD completion. It also indicates how some fields can be auto generated when they are not specified in the configuration.

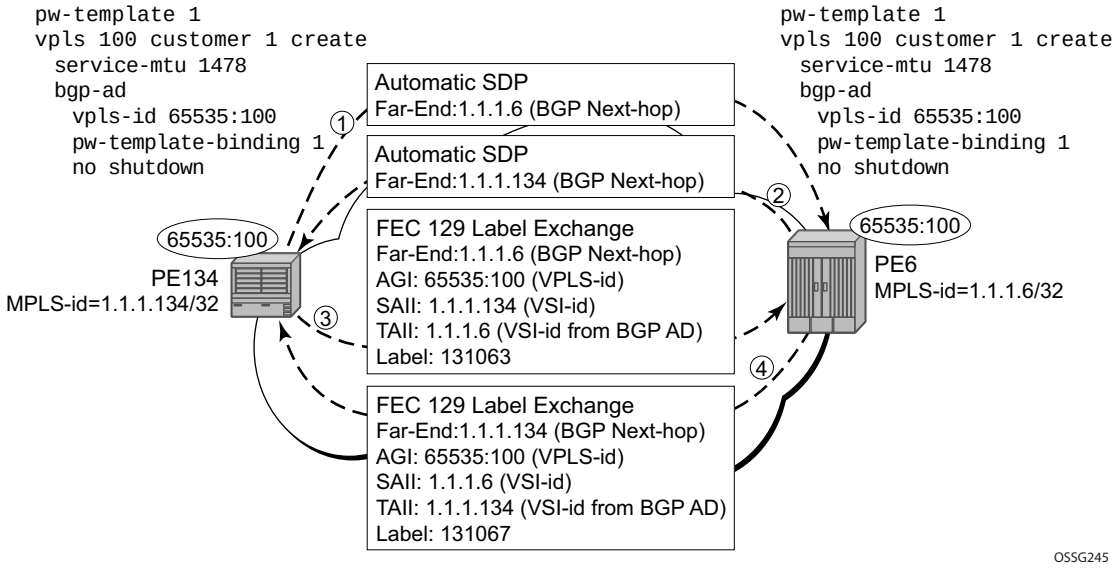

Figure 43: BGP AD Triggering LDP Functions

The first command will display the LDP peering relationships that have been established (Figure 44). The type of adjacency is displayed in the “Adj Type” column. In this case the type is “Both” meaning link and targeted sessions have been successfully established.

```
PERs6# show router ldp session
LDP Sessions
Peer LDP Id         Adj Type   State         Msg Sent  Msg Recv  Up Time
1.1.1.134:0         Both       Established   21482     21482     0d 15:38:44
No. of Sessions: 1
```

Figure 44: Show Router LDP Session Output

The second command shows the specific LDP service label information broken up per FEC element type, 128 or 129, basis (Figure 45). The information for FEC element 129 includes the AGI, SAII and the TAII.

```
PERs6# show router ldp bindings fec-type services
LDP LSR ID: 1.1.1.6
Legend: U - Label In Use, N - Label Not In Use, W - Label Withdrawn
        S - Status Signaled Up,  D - Status Signaled Down
        E - Epipe Service, V - VPLS Service, M - Mirror Service
        A - Apipe Service, F - Fpipe Service, I - IES Service, R - VPRN service
        P - Ipipe Service, C - Cpipe Service
        TLV - (Type, Length: Value)
LDP Service FEC 128 Bindings
Type   VCId       SvcId      SDPId  Peer            IngLbl  EgrLbl  LMTU  RMTU
No Matching Entries Found

LDP Service FEC 129 Bindings
AGI                                SAII                    TAII
Type               SvcId      SDPId  Peer            IngLbl  EgrLbl  LMTU  RMTU
65535:100                          1.1.1.6                 1.1.1.134
V-Eth              100        17406  1.1.1.134       131063U 131067S 1464  1464
No. of FEC 129s: 1
```

Figure 45: Show Router LDP Bindings FEC-Type Services

Pseudowire Template

The pw-template is defined under the top level service command (**config>service# pw-template**) and specifies whether to use an automatically generated SDP or manually configured SDP. It also provides the set of parameters required for establishing the pseudowire (SDP binding) as displayed in Figure 46.

```
PERs6>config>service# pw-template 1 create
  - [no] pw-template <policy-id> [use-provisioned-sdp]

 <policy-id>          : [1..2147483647]
 <use-provisioned-s*> : keyword


 [no] accounting-pol* - Configure accounting-policy to be used
 [no] collect-stats   - Enable/disable statistics collection
 [no] disable-aging   - Enable/disable aging of MAC addresses
 [no] disable-learni* - Enable/disable learning of new MAC addresses
 [no] discard-unknow* - Enable/disable discarding of frames with unknown
                        source MAC address
      egress          + Spoke SDP binding egress configuration
      igmp-snooping   + Configure IGMP snooping parameters
      ingress         + Spoke SDP binding ingress configuration
 [no] limit-mac-move  - Configure mac move
 [no] mac-pinning     - Enable/disable MAC address pinning on this spoke SDP
 [no] max-nbr-mac-ad* - Configure the maximum number of MAC entries in the FDB
                        from this SDP
 [no] split-horizon-* + Configure a split horizon group
      vc-type         - Configure VC type
 [no] vlan-vc-tag     - Configure VLAN VC tag
```

Figure 46: PW-Template CLI Tree

A **pw-template-binding** command configured within the VPLS service under the **bgp-ad** sub-command is a pointer to the pw-template that should be used. If a VPLS service does not specify an import-rt list, then that binding applies to all route targets accepted by that VPLS. The **pw-template-bind** command can select a different template on a per import-rt basis. It is also possible to specify specific pw-templates for some route targets with a VPLS service and use the single **pw-template-binding** command to address all unspecified but accepted imported targets.

```
PERs6>config>service>vpls>bgp-ad# pw-template-binding
  - pw-template-binding <policy-id> [split-hozion-group <group-name>] [import-
rt
    {ext-community, ...(upto 5 max)}]
  - no pw-template-binding <policy-id>

 <policy-id>          : [1..2147483647]
 <group-name>         : [32 chars max]
 <ext-community>      : target:{<ip-addr:comm-val>|<as-number:ext-comm-val>}
                        ip-addr      - a.b.c.d
                        comm-val     - [0..65535]
                        as-number    - [1..65535]
                        ext-comm-val - [0..4294967295]
```

Figure 47: PW-Template-Binding CLI Syntax

It is important understand the significance of the split-horizon-group used by the pw-template. Traditionally, when a VPLS instance was manually created using mesh-sdp bindings, these were automatically placed in a common split-horizon-group to prevent forwarding between the pseudowire in the VPLS instances. This prevents loops that would have otherwise occurred in the Layer 2 service. When automatically discovering VPLS service using BGP AD the service provider has the option of associating the auto-discovered pseudowire with a split-horizon group to control the forwarding between pseudowires.

Configuring BGP VPLS

This section gives a configuration example required to bring up BGP VPLS in the VPLS PEs depicted in Figure 48:

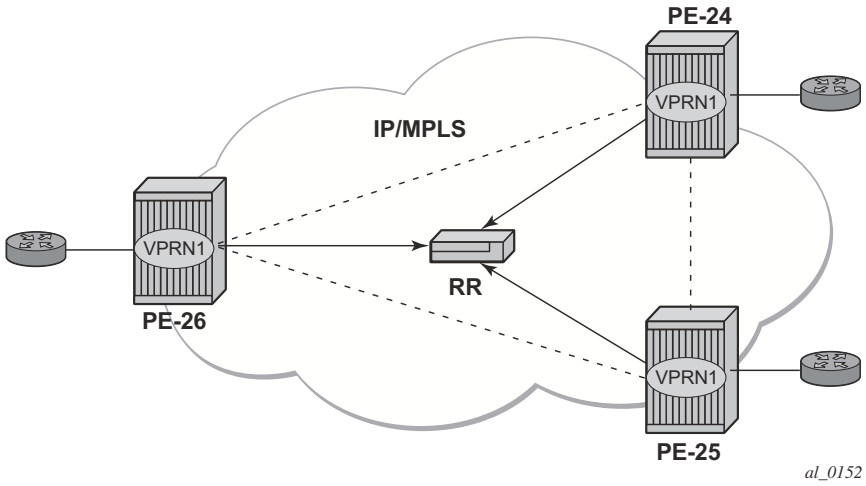

Figure 48: BGP VPLS Example

The red BGP VPLS is configured in the PE24, PE25 and PE26 using the commands shown in the following CLI examples.

```
*A:PE24>config>service>vpls# info
----------------------------------------------
            bgp
                route-distinguisher 65024:600
                route-target export target:65019:600 import target:65019:600
                pw-template-binding 1
            exit
            bgp-vpls
                max-ve-id 100
                ve-name 24
                    ve-id 24
                exit
                no shutdown
            exit
            sap 1/1/20:600.* create
            exit
            no shutdown
----------------------------------------------
*A:PE24>config>service>vpls#


*A:PE25>config>service>vpls# info
----------------------------------------------
            bgp
                route-distinguisher 65025:600
                route-target export target:65019:600 import target:65019:600
```

```
                pw-template-binding 1
            exit
            bgp-vpls
                max-ve-id 100
                ve-name 25
                    ve-id 25
                exit
                no shutdown
            exit
            sap 1/1/19:600.* create
            exit
            no shutdown
-----------------------------------------------
*A:PE25>config>service>vpls#


*A:PE26>config>service>vpls# info
-----------------------------------------------
            bgp
                route-distinguisher 65026:600
                route-target export target:65019:600 import target:65019:600
                pw-template-binding 1
            exit
            bgp-vpls
                max-ve-id 100
                ve-name 26
                    ve-id 26
                exit
                no shutdown
            exit
            sap 5/2/20:600.* create
            exit
            no shutdown
-----------------------------------------------
*A:PE26>config>service>vpls#
```

Configuring Provider Edge Discovery Policies

Use the following CLI syntax to create PE discovery policy.

CLI Syntax:
```
config>service# pe-discovery-policy name
                password password
                polling-interval minutes
                server server-index address ip-address secret key
                   [hash|hash2] [port port-num]
                timeout seconds
```

The following displays the PE discovery policy configuration.

```
A:ALA-48>config>service# info
----------------------------------------------
        pe-discovery-policy "RAD_Disc for Service 103"
            password "timetravpn"
            polling-interval 1
            timeout 10
            server 1 address 192.168.15.125 secret "LwyBQX4E2C/bXAGTtpNeYk" hash2 port 1812
          server 2 address 192.168.15.122 secret "cj0n8F.5UUl5WBegZ.m6WmvwTYw6MZu0" hash2 port 1812
        exit
        customer 1 create
            description "Default customer"
        exit
...
----------------------------------------------
A:ALA-48
```

Configuring a VPLS Management Interface

Use the following CLI syntax to create a VPLS management interface.

CLI Syntax:
```
config>service>vpls# interface ip-int-name
address ip-address[/mask] [netmask]
arp-timeout seconds
description description-string
mac ieee-address
no shutdown
static-arp ip-address ieee-address
```

The following displays the configuration.

```
A:ALA-49>config>service>vpls>interface# info detail
----------------------------------------------
              no description
              mac 14:31:ff:00:00:00
              address 123.231.10.10/24
              no arp-timeout
              no shutdown
----------------------------------------------
A:ALA-49>config>service>vpls>interface#
```

Applying a PE Discovery Policy to a VPLS Service

Use the following CLI syntax to PE discovery parameters to a VPLS service.

CLI Syntax:
```
config>service# vpls service-id
             radius-discovery
                pe-discovery-policy name
                no shutdown
                user-name-format {vpn-id vpn-id | router-distinguisher rd}
             sap sap-id [split-horizon-group group-name]
                description description-string
             split-horizon-group group-name>[residential-group]
                restrict-protected-src [alarm-only]
```

The following displays a VPLS PE discovery policy configuration.

```
*A:ALA-48>config>service>vpls# info
----------------------------------------------
            description "Default sap description for service id 103"
            split-horizon-group "SHG-RAD_Disc" create
                restrict-protected-src
            exit
            stp
                no shutdown
            exit
            radius-discovery
                pe-discovery-policy "RAD_Disc for Service 103"
                user-name-format vpn-id 901:103
                no shutdown
            exit
            sap 1/1/7:0 create
                description "Default sap description for service id 103"
                static-mac 12:34:56:78:90:0f create
            exit
            sap 1/1/11:0 split-horizon-group "SHG-RAD_Disc" create
                description "SHG for Radius Discovery"
            exit
            no shutdown
----------------------------------------------
*A:ALA-48>config>service>vpls#
```

Configuring Policy-Based Forwarding for Deep Packet Inspection (DPI) in VPLS

The purpose of policy-based forwarding is to capture traffic from a customer and perform a deep packet inspection (DPI) and forward traffic, if allowed, by the DPI.

In the following example, the split horizon groups are used to prevent flooding of traffic. Traffic from customers enter at SAP 1/1/5:5. Due to the mac-filter 100 that is applied on ingress, all traffic with dot1p 07 marking will be forwarded to SAP 1/1/22:1, which is the DPI.

DPI performs packet inspection/modification and either drops the traffic or forwards the traffic back into the box through SAP 1/1/21:1. Traffic will then be sent to spoke-sdp 3:5.

SAP 1/1/23:5 is configured to see if the VPLS service is flooding all the traffic. If flooding is performed by the router then traffic would also be sent to SAP 1/1/23:5 (which it should not).

Figure 49 shows an example to configure policy-based forwarding for deep packet inspection on a VPLS service. For information about configuring filter policies, refer to the 7750 SR OS Router Configuration Guide.

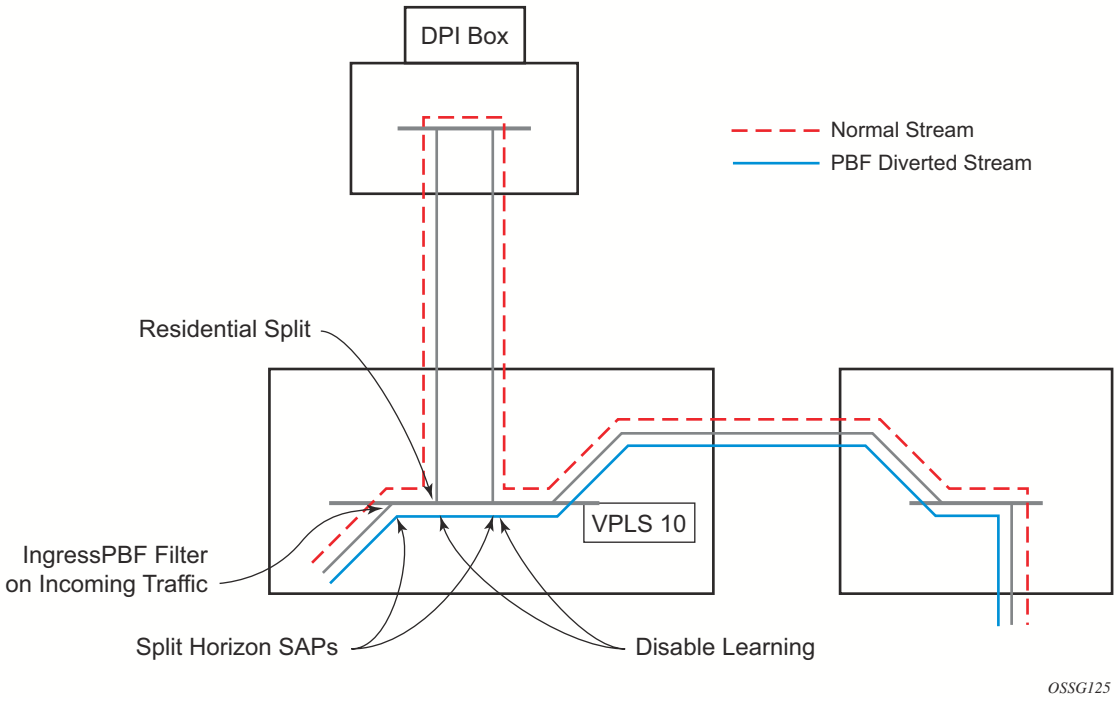

Figure 49: Policy-Based Forwarding For Deep Packet Inspection

The following example displays the service configuration:

```
*A:ALA-48>config>service# info
----------------------------------------------
...
        vpls 10 customer 1 create
            service-mtu 1400
            split-horizon-group "dpi" residential-group create
            exit
            split-horizon-group "split" create
            exit
            stp
                shutdown
            exit
            igmp-host-tracking
                expiry-time 65535
                no shutdown
            exit
            sap 1/1/21:1 split-horizon-group "split" create
                disable-learning
                static-mac 00:00:00:31:11:01 create
            exit
            sap 1/1/22:1 split-horizon-group "dpi" create
                disable-learning
                static-mac 00:00:00:31:12:01 create
            exit
            sap 1/1/23:5 create
                static-mac 00:00:00:31:13:05 create
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-48>config>service#
```

The following example displays the MAC filter configuration:

```
*A:ALA-48>config>filter# info
----------------------------------------------
...
        mac-filter 100 create
            default-action forward
            entry 10 create
                match
                    dot1p 7 7
                exit
                log 101
                action forward sap 1/1/22:1
            exit
        exit
...
----------------------------------------------
*A:ALA-48>config>filter#
```

The following example displays the service configuration with a MAC filter:

```
*A:ALA-48>config>service# info
----------------------------------------------
...
        vpls 10 customer 1 create
            service-mtu 1400
            split-horizon-group "dpi" residential-group create
            exit
            split-horizon-group "split" create
            exit
            stp
                shutdown
            exit
            igmp-host-tracking
                expiry-time 65535
                no shutdown
            exit
            sap 1/1/5:5 split-horizon-group "split" create
                ingress
                    filter mac 100
                exit
                static-mac 00:00:00:31:15:05 create
            exit
            sap 1/1/21:1 split-horizon-group "split" create
                disable-learning
                static-mac 00:00:00:31:11:01 create
            exit
            sap 1/1/22:1 split-horizon-group "dpi" create
                disable-learning
                static-mac 00:00:00:31:12:01 create
            exit
            sap 1/1/23:5 create
                static-mac 00:00:00:31:13:05 create
            exit
            spoke-sdp 3:5 create
            exit
            no shutdown
        exit
....
----------------------------------------------
*A:ALA-48>config>service#
```

Service Management Tasks

This section discusses the following service management tasks:

Modifying VPLS Service Parameters

You can change existing service parameters. The changes are applied immediately.
To display a list of services, use the **show service service-using vpls** command. Enter the parameter such as description, SAP, SDP, and/or service-MTU command syntax, and then enter the new information.

The following displays a modified VPLS configuration.

```
*A:ALA-1>config>service>vpls# info
----------------------------------------------
            description "This is a different description."
            disable-learning
            disable-aging
            discard-unknown
            local-age 500
            remote-age 1000
            stp
                shutdown
            exit
            sap 1/1/5:22 create
                description "VPLS SAP"
            exit
            spoke-sdp 2:22 create
            exit
            no shutdown
----------------------------------------------
*A:ALA-1>config>service>vpls#
```

Modifying Management VPLS Parameters

To modify the range of VLANs on an access port that are to be managed by an existing management VPLS, first the new range should be entered and afterwards the old range removed. If the old range is removed before a new range is defined, all customer VPLS services in the old range will become unprotected and may be disabled.

CLI Syntax:
```
config>service# vpls service-id
    sap sap-id
       managed-vlan-list
          [no] range vlan-range
```

Deleting a Management VPLS

As with normal VPLS service, a management VPLS cannot be deleted until SAPs and SDPs are unbound (deleted), interfaces are shutdown, and the service is shutdown on the service level.

Use the following CLI syntax to delete a management VPLS service:

CLI Syntax:
```
config>service
  [no] vpls service-id
     shutdown
     [no] spoke-sdp sdp-id
     [no] mesh-sdp sdp-id
        shutdown
     [no] sap sap-id
        shutdown
```

Disabling a Management VPLS

You can shut down a management VPLS without deleting the service parameters.

When a management VPLS is disabled, all associated user VPLS services are also disabled (to prevent loops). If this is not desired, first un-manage the user's VPLS service by removing them from the managed-vlan-list or moving the spoke SDPs on to another tunnel SDP.

CLI Syntax:
```
config>service
  vpls service-id
     shutdown
```

Example:
```
config>service# vpls 1
config>service>vpls# shutdown
config>service>vpls# exit
```

Deleting a VPLS Service

A VPLS service cannot be deleted until SAPs and SDPs are unbound (deleted), interfaces are shutdown, and the service is shutdown on the service level.

Use the following CLI syntax to delete a VPLS service:

CLI Syntax:
```
config>service
 [no] vpls service-id
    shutdown
    [no] mesh-sdp sdp-id
       shutdown
    sap sap-id [split-horizon-group group-name]
    no sap sap-id
       shutdown
```

Disabling a VPLS Service

You can shut down a VPLS service without deleting the service parameters.

CLI Syntax:
```
config>service> vpls service-id
 [no] shutdown
```

Example:
```
config>service# vpls 1
config>service>vpls# shutdown
config>service>vpls# exit
```

Re-Enabling a VPLS Service

To re-enable a VPLS service that was shut down.

CLI Syntax:
```
config>service> vpls service-id
  [no] shutdown
```

Example:
```
config>service# vpls 1
config>service>vpls# no shutdown
config>service>vpls# exit
```