Configuring a VLL Service with CLI

This section provides information to configure Virtual Leased Line (VLL) services using the command line interface.

Topics in this section include:

Fpipe:

Ipipe

Basic Configurations

Common Configuration Tasks

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

- Associate the service with a customer ID.
- Define SAP parameters
  - → Optional - configure ATM parameters
  - → Optional - select egress and ingress QoS and/or scheduler policies (configured in the **config>qos** context).
  - → Optional - select accounting policy (configured in the **config>log** context).
- Define spoke SDP parameters.
- Enable the service.

Configuring VLL Components

This section provides VLL configuration examples for the VLL services:

Creating an Apipe Service

Use the following CLI syntax to create an Apipe service.

CLI Syntax:
```
config>service# apipe service-id [customer customer-id] [vpn
  vpn-id] [vc-type {atm-vcc|atm-sdu|atm-vpc|atm-cell}][vc-switching]
        description description-string
        interworking {frf-5}
        service-mtu octets
        no shutdown
```

The following example displays the command usage to create an Apipe service:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5 customer 1 create
A:ALA-41config>service>apipe# description "apipe test"
A:ALA-41config>service>apipe# service-mtu 1400
A:ALA-41config>service>apipe# no shutdown
A:ALA-41config>service>apipe#
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# apipe 5 customer 1 create
A:ALA-42>config>service>apipe# description "apipe test"
A:ALA-42>config>service>apipe# service-mtu 1400
A:ALA-42>config>service>apipe# no shutdown
A:ALA-42>config>service>apipe#
```

The following example displays the Apipe service creation output.

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            no shutdown
        exit
...
------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            no shutdown
        exit
...
------------------------------------
A:ALA-42>config>service#
```

Configuring Basic Apipe SAP Parameters

Use the following CLI syntax to configure Apipe SAP parameters.

CLI Syntax:
```
config>service# apipe service-id [customer customer-id] [vpn
  vpn-id] [vc-type {atm-vcc|atm-sdu|atm-vpc|atm-cell}][vc-switching]
           sap sap-id
              accounting-policy acct-policy-id
              atm
                 egress
                    traffic-desc traffic-desc-profile-id
                 ingress
                    traffic-desc traffic-desc-profile-id
                 oam
                    alarm-cells
                    terminate
              collect-stats
              description description-string
              egress
                 qos policy-id
                 scheduler-policy scheduler-policy-name
              ingress
                 qos policy-id [shared-queuing]
                 scheduler-policy scheduler-policy-name
              multi-service-site customer-site-name
              no shutdown
```

The following example displays the command usage to create Apipe SAPs:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# sap 1/1/1:0/32 create
A:ALA-41>config>service>apipe>sap# ingress
A:ALA-41>config>service>apipe>sap>ingress# qos 102
A:ALA-41>config>service>apipe>sap>ingress# exit
A:ALA-41>config>service>apipe>sap# egress
A:ALA-41>config>service>apipe>sap>egress# qos 103
A:ALA-41>config>service>apipe>sap>egress# exit
A:ALA-41>config>service>apipe>sap# no shutdown
A:ALA-41>config>service>apipe>sap# exit
A:ALA-41>config>service>apipe#
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# apipe 5
A:ALA-42>config>service>apipe# sap 2/2/2:0/32 create
A:ALA-42>config>service>apipe>sap# ingress
A:ALA-42>config>service>apipe>sap>ingress# qos 102
```

```
A:ALA-42>config>service>apipe>sap>ingress# exit
A:ALA-42>config>service>apipe>sap# egress
A:ALA-42>config>service>apipe>sap>egress# qos 103
A:ALA-42>config>service>apipe>sap>egress# exit
A:ALA-42>config>service>apipe>sap# no shutdown
A:ALA-42>config>service>apipe>sap# exit
A:ALA-42>config>service>apipe#
```

The following output displays the Apipe SAP configuration.

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            sap 1/1/1:0/32 create
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            sap 2/2/2:0/32 create
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-42>config>service#
```

Configuring an ATM SAP in the N-to-1 Mapping of ATM VPI/VCI to ATM Pseudowire

Users can configure an ATM-cell Apipe service with a new ATM SAP type. The SAP type refers to a pre-configured ATM connection profile name.

```
configure service apipe 100 vc-type atm-cell
      sap <port-id|aps-id>[:cp.<connection-profile-num>]
```

The ATM SAP connection profile is configured with the list of discrete VPI/VCI values.

```
configure connection-profile 2 {member vpi/vci...(up to 16)}
```

A connection profile can only be applied to a SAP which is part of an apipe VLL service of vc-type atm-cell. The ATM SAP can be on a regular port or APS port. A connection profile can be applied to any number of ATM SAPs.

Up to a maximum of 16 discrete VPI/VCI values can be configured in a connection profile. After creation of the connection profile, the user can subsequently add, remove, or modify the VPI/VCI entries. This triggers a re-evaluation of all the ATM SAPs which are currently using that profile.

The user must also override the PW type signaled to type '0x0009 N:1 VCC cell' by using the following command:

```
configure>service>apipe>signaled-vc-type-override atm-vcc
```

This command is not allowed in an Apipe VLL of vc-type value atm-cell if a configured ATM SAP is not using a connection profile. Conversely, if the signaling override command is enabled, only an ATM SAP with a connection profile assigned will be allowed.

The override command is not allowed on an Apipe VLL service of vc-type value other than atm-cell. It is also not allowed on a VLL service with the vc-switching option enabled since signaling of the pseudowire FEC in a Multi-Segment Pseudowire (MS-PW) is controlled by the T-PE nodes. Thus for this feature to be used on a MS-PW, it is required to configure an Apipe service of vc-type atm-cell at the T-PE nodes with the **signaled-vc-type-override** command enabled, and to configure an Apipe VLL service of vc-type atm-vcc at the S-PE node with the vc-switching option enabled.

The following are the restrictions of this feature:

- A SAP-to-SAP VLL service is not supported using ATM SAP with a connection profile assigned. The user must configure each VPI/VCI into a separate SAP and create as many Apipe VLL services of type atm-vcc as required.
- An ATM SAP with a connection profile assigned cannot be configured on a port with is part of a MC-APS protection group.
- It is strongly recommended to not apply a VCI based QoS Filter to the ingress of an ATM SAP with a connection profile. Because the filter matches the VCI value of the first cell of a concatenated packet, the entire packet will be treated the same way based on the action of the entry of the criteria, all cells of the concatenated packet are mapped to the same FC and profile based on the VCI value of the first cell.

This feature is supported on the 4-port OC-3/STM-1:OC-12/STM-4 ATM MDA and on the 16-port OC-3/STM-1 ATM MDA and is supported IOM3/IMM and in any chassis mode on the 7750 SR-7, and 7750 SR-12 as well as the 7750-C4 and C12 chassis.

Configuring Apipe SDP Bindings

Use the following CLI syntax to create a spoke SDP binding with an Apipe service:

CLI Syntax:
```
config>service# apipe service-id [customer customer-id] [vpn
  vpn-id] [vc-type {atm-vcc|atm-sdu|atm-vpc|atm-cell}] [vc-switching]
       spoke-sdp sdp-id:vc-id
          cell-concatenation
             aal5-frame-aware
             clp-change
             max-cells cell-count
             max-delay delay-time
          egress
             vc-label egress-vc-label
          ingress
             vc-label ingress-vc-label
          no shutdown
```

The following example displays the command usage to create Apipe spoke SDPs:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# spoke-sdp 1:5 create
A:ALA-41>config>service>apipe>spoke-sdp# no shutdown
A:ALA-41>config>service>apipe>spoke-sdp# exit
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# apipe 5
A:ALA-42>config>service>apipe# spoke-sdp 1:5 create
A:ALA-42>config>service>apipe>spoke-sdp# no shutdown
A:ALA-42>config>service>apipe>spoke-sdp# exit
```

The following output displays the Apipe spoke SDP configurations.

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            sap 1/1/1:0/32 create
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
```

```
            spoke-sdp 1:5 create
            exit
            no shutdown
        exit
...
-----------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
-----------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            sap 2/2/2:0/32 create
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
            spoke-sdp 1:5 create
            exit
            no shutdown
        exit
...
-----------------------------------
A:ALA-42>config>service#
```

Creating a Cpipe Service

Basic Configuration

Use the following CLI syntax to create a Cpipe service. A route distinguisher must be defined in order for Cpipe to be operationally active.

CLI Syntax: `config>service# cpipe service-id [customer customer-id] [vpn vpn-id] [vc-type {satop-e1 | satop-t1 | cesopsn | cesopsn-cas}] [vc-switching] [create]`

The following displays a Cpipe service configuration example.

```
*A:ALA-1>config>service# info
----------------------------------------------
...
        cpipe 210 customer 1 vc-type cesopsn create
            service-mtu 1400
            sap 1/5/1.1.3.1 create
            exit
            spoke-sdp 1:210 create
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-1>config>service#
```

Configuration Requirements

Before a Cpipe service can be provisioned, the following tasks must be completed:

- Configuring a DS1 Port on page 140
- Configuring a Channel Group on page 141

Configuring a DS1 Port

The following displays an example of a DS1 port configured for CES.

```
A:sim216#  show port 1/5/1.1.3.1
===============================================================================
TDM DS1 Interface
===============================================================================
Description        : DS1
Interface          : 1/5/1.1.3,1
Type               : ds1                       Framing             : esf
Admin Status       : up                        Oper Status         : up
Physical Link      : yes                       Clock Source        : loop-timed
Signal Mode        : none
Last State Change  : 10/31/2006 14:23:12       Channel IfIndex     : 580943939
Loopback           : none                      Invert Data         : false
Remote Loop respond: false                     In Remote Loop      : false
Load-balance-algo  : default                   Egr. Sched. Pol     : n/a
BERT Duration      : N/A                       BERT Pattern        : none
BERT Synched       : 00h00m00s                 Err Insertion Rate  : 0
BERT Errors        : 0                         BERT Status         : idle
BERT Total Bits    : 0
Cfg Alarm          : ais los
Alarm Status       :
===============================================================================
A:sim216#
```

Configuring a Channel Group

The following displays an example of a DS1 channel group configured for CES.

```
A:sim216#  show port 1/5/1.1.3.1
===============================================================================
TDM DS0 Chan Group
===============================================================================
Description        : DS0GRP
Interface          : 1/5/1.1.3.1
TimeSlots          : 1-12
Speed              : 64                      CRC                 : 16
Admin Status       : up                      Oper Status         : up
Last State Change  : 10/31/2006 14:23:12     Chan-Grp IfIndex    : 580943940
Configured mode    : access                  Encap Type          : cem
Admin MTU          : 4112                    Oper MTU            : 4112
Physical Link      : Yes                     Bundle Number       : none
Idle Cycle Flags   : flags                   Load-balance-algo   : default
Egr. Sched. Pol    : n/a
===============================================================================
A:sim216#
```

Configuring Cpipe SAPs and Spoke SDPs

The following output displays examples of Cpipe SAP and spoke SDP configurations.

```
*A:ALA-49>config>service# info
#------------------------------------------------
echo "Service Configuration"
#------------------------------------------------
...
        cpipe 100 customer 1 vc-type cesopsn create
            service-mtu 1400
            sap 1/5/1.1.1.1 create
            exit
            spoke-sdp 1:100 create
            exit
            no shutdown
        exit
        cpipe 200 customer 1 vc-type cesopsn-cas create
            sap 1/5/1.2.1.1 create
            exit
            sap 1/5/1.2.2.1 create
            exit
            no shutdown
        exit
        cpipe 210 customer 1 vc-type cesopsn-cas create
            service-mtu 1400
            sap 1/5/1.1.3.1 create
            exit
            spoke-sdp 1:210 create
            exit
            no shutdown
        exit
        cpipe 300 customer 1 vc-type cesopsn create
            sap 1/5/1.3.4.1 create
            exit
            sap 1/5/1.3.6.1 create
            exit
            no shutdown
        exit
        cpipe 400 customer 1 vc-type satop-e1 create
            sap 1/5/1.2.3.1 create
            exit
            spoke-sdp 1:400 create
            exit
            no shutdown
        exit
...
#------------------------------------------------
*A:ALA-49>config>service#
```

```
A:sim213>config>service>cpipe# info
----------------------------------------------
            description "cpipe-100"
            sap 1/5/1.1.1.1 create
                cem
                    packet jitter-buffer 16 payload-size 384
                    report-alarm rpktloss
                    no report-alarm stray
                    rtp-header
                exit
            exit
            spoke-sdp 1:100 create
            exit
            no shutdown
----------------------------------------------
A:sim213>config>service>cpipe#
```

Creating an Epipe Service

Use the following CLI syntax to create an Epipe service.

CLI Syntax: `config>service# epipe service-id [customer customer-id] [vpn vpn-id] [vc-switching]`
`description description-string`
`no shutdown`

The following displays an Epipe configuration example:

```
A:ALA-1>config>service# info
----------------------------------------
...
        epipe 500 customer 5 vpn 500 create
            description "Local epipe service"
            no shutdown
        exit
----------------------------------------
A:ALA-1>config>service#
```

Configuring Epipe SAP Parameters

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

Use the following CLI syntax to create:

- Local Epipe SAPs on page 146
- Distributed Epipe SAPs on page 148

CLI Syntax:

```
config>service# epipe service-id [customer customer-id]
        sap sap-id [endpoint endpoint-name]
        sap sap-id [no-endpoint]
              accounting-policy policy-id
              collect-stats
              description description-string
              no shutdown
              egress
                 filter {ip ip-filter-name | mac mac-filter-name}
                 qos sap-egress-policy-id
                 scheduler-policy scheduler-policy-name
              ingress
                 filter {ip ip-filter-name | mac mac-filter-name}
                 match-qinq-dot1p {top|bottom}
                 qos policy-id [shared-queuing]
                 scheduler-policy scheduler-policy-name
```

Local Epipe SAPs

Table 5: Supported SAP Types

Uplink Type	Svc SAP Type	Cust. VID	Access SAPs	Network SAPs
L2	Null-star	N/A	Null, dot1q *	Q.*
L2	Dot1q	N/A	Dot1q	Q.*
L2	Dot1q-preserve	X	Dot1q (encap = X)	Q1.Q2 (where Q2 = X)

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

By default, QoS policy ID 1 is applied to ingress and egress service SAPS. Existing filter policies or other existing QoS policies can be associated with service SAPs on ingress and egress ports.

An existing scheduler policy can be applied to ingress and egress SAPs to be used by the SAP queues. The schedulers comprising the policy are created at the time the scheduler policy is applied to the SAP. If any orphaned queues (queues with a non-existent local scheduler defined) exist on a SAP and the policy application creates the required scheduler, the status on the queue becomes non-orphaned at this time.

Ingress and Egress SAP parameters can be applied to local and distributed Epipe service SAPs.

This example displays the SAP configurations for local Epipe service 500 on SAP 1/1/2 and SAP 1/1/3 on ALA-1.

```
A:ALA-1>config>service# epipe 500 customer 5 create
     config>service>epipe$ description "Local epipe service"
     config>service>epipe# sap 1/1/2:0 create
     config>service>epipe>sap? ingress
     config>service>epipe>sap>ingress# qos 20
     config>service>epipe>sap>ingress# filter ip 1
     config>service>epipe>sap>ingress# exit
     config>service>epipe>sap# egress
     config>service>epipe>sap>egress# qos 20
     config>service>epipe>sap>egress# scheduler-policy test1
     config>service>epipe>sap>egress# exit
     config>service>epipe>sap# no shutdown
     config>service>epipe>sap# exit

     config>service>epipe# sap 1/1/3:0 create
     config>service>epipe>sap# ingress
     config>service>epipe>sap>ingress# qos 555
     config>service>epipe>sap>ingress# filter ip 1
     config>service>epipe>sap>ingress# exit
     config>service>epipe>sap# egress
     config>service>epipe>sap>egress# qos 627
     config>service>epipe>sap>egress# scheduler-policy alpha
     config>service>epipe>sap>egress# exit
     config>service>epipe>sap# no shutdown
     config>service>epipe>sap# exit
```

The following example displays the local Epipe configuration:

```
A:ALA-1>config>service# info
----------------------------------------------
...
        epipe 500 customer 5 vpn 500 create
            description "Local epipe service"
            sap 1/1/2:0 create
                ingress
                    qos 20
                    filter ip 1
                exit
                egress
                    scheduler-policy "test1"
                    qos 20
                exit
            exit
            sap 1/1/3:0 create
                ingress
                    qos 555
                    filter ip 1
                exit
                egress
                    scheduler-policy "alpha"
                    qos 627
                exit
            exit
            no shutdown
        exit
----------------------------------------------
A:ALA-1>config>service#
```

Distributed Epipe SAPs

To configure a distributed Epipe service, you must configure service entities on the originating and far-end nodes. You should use the same service ID on both ends (for example, Epipe 5500 on ALA-1 and Epipe 5500 on ALA-2). The **spoke-sdp** *sdp-id:vc-id* must match on both sides. A distributed Epipe consists of two SAPs on different nodes.

By default, QoS policy ID 1 is applied to ingress and egress service SAPS. Existing filter policies or other existing QoS policies can be associated with service SAPs on ingress and egress.

An existing scheduler policy can be applied to ingress and egress SAPs to be used by the SAP queues. The schedulers comprising the policy are created at the time the scheduler policy is applied to the SAP. If any orphaned queues (queues with a non-existent local scheduler defined) exist on a SAP and the policy application creates the required scheduler, the status on the queue becomes non-orphaned at this time.

Ingress and egress SAP parameters can be applied to local and distributed Epipe service SAPs.

For SDP configuration information, see the *Services Overview Guide*. For SDP binding information, see Configuring SDP Bindings on page 152.

This example configures a distributed service between ALA-1 and ALA-2.

```
A:ALA-1>epipe 5500 customer 5 create
     config>service>epipe$ description "Distributed epipe service to east coast"
     config>service>epipe# sap 221/1/3:21 create
     config>service>epipe>sap# ingress
     config>service>epipe>sap>ingress# qos 555
     config>service>epipe>sap>ingress# filter ip 1
     config>service>epipe>sap>ingress# exit
     config>service>epipe>sap# egress
     config>service>epipe>sap>egress# qos 627
     config>service>epipe>sap>egress# scheduler-policy alpha
     config>service>epipe>sap>egress# exit
     config>service>epipe>sap# no shutdown
     config>service>epipe>sap# exit
     config>service>epipe#

A:ALA-2>config>service# epipe 5500 customer 5 create
     config>service>epipe$ description "Distributed epipe service to west coast"
     config>service>epipe# sap 441/1/4:550 create
     config>service>epipe>sap# ingress
     config>service>epipe>sap>ingress# qos 654
     config>service>epipe>sap>ingress# filter ip 1020
     config>service>epipe>sap>ingress# exit
     config>service>epipe>sap# egress
     config>service>epipe>sap>egress# qos 432
     config>service>epipe>sap>egress# filter ip 6
     config>service>epipe>sap>egress# scheduler-policy test1
     config>service>epipe>sap>egress# exit
     config>service>epipe>sap# no shutdown
     config>service>epipe#
```

The following example displays the SAP configurations for ALA-1 and ALA-2:

```
A:ALA-1>config>service# info
----------------------------------------------
...
        epipe 5500 customer 5 vpn 5500 create
            description "Distributed epipe service to east coast"
            sap 221/1/3:21 create
                ingress
                    qos 555
                    filter ip 1
                exit
                egress
                    scheduler-policy "alpha"
                    qos 627
                exit
            exit
        exit
...
----------------------------------------------
A:ALA-1>config>service#


A:ALA-2>config>service# info
----------------------------------------------
...
        epipe 5500 customer 5 vpn 5500 create
            description "Distributed epipe service to west coast"
            sap 441/1/4:550 create
                ingress
                    qos 654
                    filter ip 1020
                exit
                egress
                    scheduler-policy "test1"
                    qos 432
                    filter ip 6
                exit
            exit
        exit
...
----------------------------------------------
A:ALA-2>config>service#
```

PBB Epipe Configuration

The following example displays the PBB Epipe configuration:

```
*A:Wales-1>config>service>epipe# info
----------------------------------------------------------------------
...
     description "Default epipe description for service id 20000"
     pbb-tunnel 200 backbone-dest-mac 00:03:fa:15:d3:a8 isid 20000
     sap 1/1/2:1.1 create
          description "Default sap description for service id 20000"
          ingress
          filter mac 1
          exit
     exit
     no shutdown
----------------------------------------------------------------------
*A:Wales-1>config>service>epipe#
```

CLI Syntax: configure service vpls 200 customer 1 b-vpls create

```
*A:Wales-1>config>service>vpls# info
----------------------------------------------------------------------
...
     service-mtu 2000
     fdb-table-size 131071
     stp
     no shutdown
     exit
     sap 1/1/8 create
     exit
     sap 1/2/3:200 create
     exit
     mesh-sdp 1:200 create
     exit
     mesh-sdp 100:200 create
     exit
     mesh-sdp 150:200 create
     exit
     mesh-sdp 500:200 create
     exit
     no shutdown
----------------------------------------------------------------------
*A:Wales-1>config>service>vpls#
```

Configuring Ingress and Egress SAP Parameters

By default, QoS policy ID 1 is applied to ingress and egress service SAPs. Existing filter policies or other existing QoS policies can be associated with service SAPs on ingress and egress ports.

An existing scheduler policy can be applied to ingress and egress SAPs to be used by the SAP queues. The schedulers comprising the policy are created at the time the scheduler policy is applied to the SAP. If any orphaned queues (queues with a non-existent local scheduler defined) exist on a SAP and the policy application creates the required scheduler, the status on the queue becomes non-orphaned at this time.

Ingress and egress SAP parameters can be applied to local and distributed Epipe service SAPs.

This example displays SAP ingress and egress parameters.

```
ALA-1>config>service# epipe 5500
     config>service>epipe# sap 2/1/3:21
     config>service>epipe>sap# ingress
     config>service>epipe>sap>ingress# qos 555
     config>service>epipe>sap>ingress# filter ip 1
     config>service>epipe>sap>ingress# exit
     config>service>epipe>sap# egress
     config>service>epipe>sap>egress# qos 627
     config>service>epipe>sap>egress# scheduler-policy alpha
     config>service>epipe>sap>egress# exit
     config>service>epipe>sap#
```

The following example displays the Epipe SAP ingress and egress configuration:

```
A:ALA-1>config>service#
-------------------------------------------
...
        epipe 5500 customer 5 vpn 5500 create
            description "Distributed epipe service to east coast"
            sap 2/1/3:21 create
                ingress
                    qos 555
                    filter ip 1
                exit
                egress
                    scheduler-policy "alpha"
                    qos 627
                exit
            exit
            spoke-sdp 2:123 create
                ingress
                    vc-label 6600
                exit
                egress
                    vc-label 5500
                exit
            exit
            no shutdown
        exit
-------------------------------------------
A:ALA-1>config>service#
```

Configuring SDP Bindings

Figure 37 displays an example of a distributed Epipe service configuration between two routers, identifying the service and customer IDs, and the uni-directional SDPs required to communicate to the far-end routers.

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.

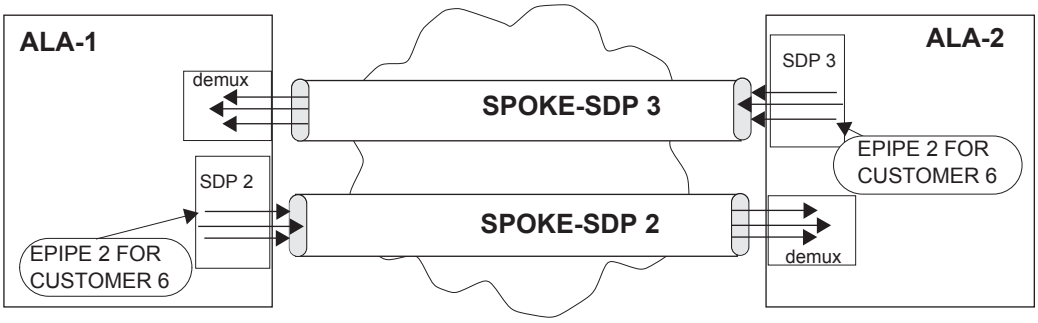

Figure 37: SDPs — Uni-Directional Tunnels

Use the following CLI syntax to create a spoke SDP binding with an Epipe service:

CLI Syntax:
```
config>service# epipe service-id [customer customer-id]
    spoke-sdp sdp-id:vc-id [vc-type {ether | vlan}]
        vlan-vc-tag 0..4094
        egress
            filter {ip ip-filter-id}
            vc-label egress-vc-label
        ingress
            filter {ip ip-filter-id}
            vc-label ingress-vc-label
        no shutdown
```

The following example displays the command usage to bind an Epipe service between ALA-1 and ALA-2. This example assumes the SAPs have already been configured (see Distributed Epipe SAPs on page 148).

A:ALA-1>config>service# epipe 5500

```
config>service>epipe# spoke-sdp 2:123
config>service>epipe>spoke-sdp# egress
config>service>epipe>spoke-sdp>egress# vc-label 5500
config>service>epipe>spoke-sdp>egress# exit
config>service>epipe>spoke-sdp# ingress
config>service>epipe>spoke-sdp>ingress# vc-label 6600
config>service>epipe>spoke-sdp>ingress# exit
config>service>epipe>spoke-sdp# no shutdown

ALA-2>config>service# epipe 5500
config>service>epipe# spoke-sdp 2:456
config>service>epipe>spoke-sdp# egress
config>service>epipe>spoke-sdp>egress# vc-label 6600
config>service>epipe>spoke-sdp>egress# exit
config>service>epipe>spoke-sdp# ingress
config>service>epipe>spoke-sdp>ingress# vc-label 5500
config>service>epipe>spoke-sdp>ingress# exit
config>service>epipe>spoke-sdp# no shutdown
```

This example displays the SDP binding for the Epipe service between ALA-1 and ALA-2:

```
A:ALA-1>config>service# info
----------------------------------------------
...
        epipe 5500 customer 5 vpn 5500 create
            description "Distributed epipe service to east coast"
            sap 2/1/3:21 create
                ingress
                    qos 555
                    filter ip 1
                exit
                egress
                    scheduler-policy "alpha"
                    qos 627
                exit
            exit
            spoke-sdp 2:123 create
                ingress
                    vc-label 6600
                exit
                egress
                    vc-label 5500
                exit
            exit
            no shutdown
        exit
...
----------------------------------------------
A:ALA-1>config>service#


A:ALA-2>config>service# info
----------------------------------------------
```

```
...
exit
        epipe 5500 customer 5 vpn 5500 create
            description "Distributed epipe service to west coast"
            sap 441/1/4:550 create
                ingress
                    qos 654
                    filter ip 1020
                exit
                egress
                    scheduler-policy "test1"
                    qos 432
                    filter ip 6
                exit
            exit
            spoke-sdp 2:456 create
                ingress
                    vc-label 5500
                exit
                egress
                    vc-label 6600
                exit
            exit
            no shutdown
        exit
...
-------------------------------------------------
A:ALA-2>config>service#
```

Creating an Fpipe Service

Use the following CLI syntax to create an Fpipe service.

CLI Syntax:
```
config>service# fpipe service-id [customer customer-id] [vpn
  vpn-id] [vc-type {fr-dlci}][vc-switching]
        description description-string
        service-mtu octets
        no shutdown
```

The following example displays the command usage to create an Fpipe service:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1 customer 1 create
A:ALA-41config>service>fpipe# description "fpipe test"
A:ALA-41config>service>fpipe# service-mtu 1400
A:ALA-41config>service>fpipe# no shutdown
A:ALA-41config>service>fpipe#
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# fpipe 1 customer 1 create
A:ALA-42>config>service>fpipe# description "fpipe test"
A:ALA-42>config>service>fpipe# service-mtu 1400
A:ALA-42>config>service>fpipe# no shutdown
A:ALA-42>config>service>fpipe#
```

The following example displays the Fpipe service creation output.

PE router 1 (A:ALA-41):
```
A:ALA-41>config>service# info
----------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            no shutdown
        exit
...
----------------------------------------
A:ALA-41>config>service#
```

PE router 2 (A:ALA-42):
```
A:ALA-42>config>service# info
----------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            no shutdown
        exit
...
----------------------------------------
A:ALA-42>config>service#
```

Configuring Fpipe SAP Parameters

Use the following CLI syntax to configure Fpipe SAP parameters.

CLI Syntax:
```
config>service# fpipe service-id [customer customer-id] [vpn
  vpn-id] [vc-type {fr-dlci}] [vc-switching]
           sap sap-id
              accounting-policy acct-policy-id
              collect-stats
              description description-string
              egress
                 filter [ip ip-filter-id]
                 qos policy-id
                 scheduler-policy scheduler-policy-name
              ingress
                 filter [ip ip-filter-id]
                 qos policy-id [shared-queuing]
                 scheduler-policy scheduler-policy-name
              multi-service-site customer-site-name
              no shutdown
```

The following example displays the command usage to create an Fpipe SAP:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# sap 1/2/1:16 create
A:ALA-41>config>service>fpipe>sap# ingress
A:ALA-41>config>service>fpipe>sap>ingress# qos 101
A:ALA-41>config>service>fpipe>sap>ingress# exit
A:ALA-41>config>service>fpipe>sap# egress
A:ALA-41>config>service>fpipe>sap>egress# qos 1020
A:ALA-41>config>service>fpipe>sap>egress# exit
A:ALA-41>config>service>fpipe>sap# no shutdown
A:ALA-41>config>service>fpipe>sap# exit
A:ALA-41>config>service>fpipe#
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# fpipe 1
A:ALA-42>config>service>fpipe# sap 2/1/1.1:16 create
A:ALA-42>config>service>fpipe>sap# ingress
A:ALA-42>config>service>fpipe>sap>ingress# qos 101
A:ALA-42>config>service>fpipe>sap>ingress# exit
A:ALA-42>config>service>fpipe>sap# egress
A:ALA-42>config>service>fpipe>sap>egress# qos 1020
A:ALA-42>config>service>fpipe>sap>egress# exit
A:ALA-42>config>service>fpipe>sap# no shutdown
A:ALA-42>config>service>fpipe>sap# exit
A:ALA-42>config>service>fpipe#
```

The following example displays the Fpipe SAP configurations.

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
----------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            sap 1/2/1:16 create
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            no shutdown
        exit
...
----------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
----------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            sap 2/1/1.1:16 create
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            no shutdown
        exit
...
----------------------------------------
A:ALA-42>config>service#
```

Configuring Fpipe SDP Bindings

Use the following CLI syntax to create a spoke SDP binding with an Fpipe service:

CLI Syntax:
```
config>service# fpipe service-id [customer customer-id] [vpn
   vpn-id] [vc-type {fr-dlci}][vc-switching]
         spoke-sdp sdp-id:vc-id
            egress
               filter ip ip-filter-id
               vc-label egress-vc-label
            ingress
               filter ip ip-filter-id
               vc-label ingress-vc-label
            no shutdown
```

The following example displays the command usage to create an Fpipe spoke SDP:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# spoke-sdp 1:1 create
A:ALA-41>config>service>spoke-sdp# no shutdown
A:ALA-41>config>service>spoke-sdp# exit
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# fpipe 1
A:ALA-42>config>service>fpipe# spoke-sdp 1:1 create
A:ALA-42>config>service>spoke-sdp# no shutdown
A:ALA-42>config>service>spoke-sdp# exit
```

The following output displays the Fpipe spoke SDP configuration.

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            sap 1/2/1:16 create
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            spoke-sdp 1:1 create
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            sap 2/1/1.1:16 create
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            spoke-sdp 1:1 create
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-42>config>service#
```

Creating an Ipipe Service

Use the following CLI syntax to create an Ipipe service.

CLI Syntax: config>service# ipipe *service-id* [customer *customer-id*] [vpn *vpn-id*][vc-switching]
description *description-string*
no shutdown

The following example displays an Ipipe configuration example:

```
A:ALA-1>config>service# info
----------------------------------------------
...
        ipipe 202 customer 1 create
            description "eth_ipipe"
            no shutdown
        exit
----------------------------------------------
A:ALA-1>config>service#
```

Configuring Ipipe SAP Parameters

The following displays an Ipipe SAP configuration example:

```
A:ALA-48>config>service# info
------------------------------------------------
...
        ipipe 202 customer 1 create
            sap 1/1/2:444 create
                description "eth_ipipe"
                ce-address 31.31.31.1
            exit
            sap 1/3/2:445 create
                description "eth_ipipe"
                ce-address 31.31.31.2
            exit
            no shutdown
        exit
...
------------------------------------------------
A:ALA-48>config>service#
```

The following displays a Frame Relay to Ethernet local Ipipe example:

Example:
```
config>service# ipipe 204 customer 1 create
config>service>ipipe$ sap 1/1/2:446 create
config>service>ipipe>sap$ description "eth_fr_ipipe"
config>service>ipipe>sap$ ce-address 32.32.32.1
config>service>ipipe>sap$ no shutdown
config>service>ipipe>sap$ exit
config>service>ipipe# sap 2/2/2:16 create
config>service>ipipe>sap$ ce-address 32.32.32.2
config>service>ipipe>sap$ no shutdown
config>service>ipipe>sap$ exit
config>service>ipipe# no shutdown
config>service>ipipe# exit
config>service#
```

The following displays the output:

```
A:ALA-48>config>service# info
----------------------------------------------
...
        ipipe 204 customer 1 create
            sap 1/1/2:446 create
                description "eth_fr_ipipe"
                ce-address 32.32.32.1
            exit
            sap 2/2/2:16 create
                ce-address 32.32.32.2
            exit
            no shutdown
        exit
...
----------------------------------------------
A:ALA-48>config>service#
```

The following displays a PPP to Ethernet local Ipipe example:

Example:
```
config>service# ipipe 206 customer 1 create
config>service>ipipe$ sap 1/1/2:447 create
config>service>ipipe>sap$ description "eth_ppp_ipipe"
config>service>ipipe>sap$ ce-address 33.33.33.1
config>service>ipipe>sap$ no shutdown
config>service>ipipe>sap$ exit
config>service>ipipe# sap 2/2/2 create
config>service>ipipe>sap$ description "ppp_eth_ipipe"
config>service>ipipe>sap$ ce-address 33.33.33.2
config>service>ipipe>sap$ no shutdown
config>service>ipipe>sap$ exit
config>service>ipipe# no shutdown
config>service>ipipe# exit
config>service#
```

The following displays the output:

```
A:ALA-48>config>service# info
----------------------------------------------
...
        ipipe 206 customer 1 create
            sap 1/1/2:447 create
                description "eth_ppp_ipipe"
                ce-address 33.33.33.1
            exit
            sap 2/2/2 create
                description "ppp_eth_ipipe"
                ce-address 33.33.33.2
            exit
            no shutdown
        exit
...
----------------------------------------------
A:ALA-48>config>service#
```

Configuring Ipipe SDP Bindings

The following displays an Ipipe SDP configuration example:

```
A:ALA-48>config>service# info
----------------------------------------------
...
        sdp 16 mpls create
            far-end 4.4.4.4
            ldp
            path-mtu 1600
            keep-alive
                shutdown
            exit
            no shutdown
        exit
...
        ipipe 207 customer 1 create
            shutdown
            sap 1/1/2:449 create
                description "Remote_Ipipe"
                ce-address 34.34.34.1
            exit
            spoke-sdp 16:516 create
                ce-address 31.31.31.2
            exit
        exit
...

----------------------------------------------
A:ALA-48>config>service#
```

Using Spoke SDP Control Words

The control word command provides the option to add a control word as part of the packet encapsulation for PW types for which the control word is optional. These are Ethernet pseudowire (Epipe), ATM N:1 cell mode pseudowires (Apipe vc-types atm-vcc and atm-vpc) and VT pseudowire (Apipe vc-type atm-cell). The control word might be needed because when ECMP is enabled on the network, packets of a given pseudowire may be spread over multiple ECMP paths if the hashing router mistakes the PW packet payload for an IPv4 or IPv6 packet. This occurs when the first nibble following the service label corresponds to a value of 4 or 6.

The control word negotiation procedures described in Section 6.2 of RFC 4447 are not supported and therefore the service will only come up if the same C bit value is signaled in both directions. If a spoke-sdp is configured to use the control word but the node receives a label mapping message with a C-bit clear, the node releases the label with an “Illegal C-bit” status code per Section 6.1 of RFC 4447. As soon as the user enables control of the remote peer, the remote peer withdraws its original label and sends a label mapping with the C-bit set to 1 and the VLL service is up in both nodes.

When the control word is enabled, VCCV packets also include the VCCV control word. In that case, the VCCV CC type 1 (OAM CW) is signaled in the VCCV parameter in the FEC. If the control word is disabled on the spoke-sdp, then the Router Alert label is used. In that case, VCCV CC type 2 is signaled. Note that for a multi-segment pseudowire (MS-PW), the CC type 1 is the only supported and thus the control word must be enabled on the spoke-sdp to be able to use VCCV-ping and VCCV-trace.

The following displays a spoke SDP control word configuration example:

```
-Dut-B>config>service>epipe# info
----------------------------------------------
     description "Default epipe description for service id 2100"
     sap 1/2/7:4 create
          description "Default sap description for service id 2100"
     exit
     spoke-sdp 1:2001 create
          control-word
     exit
     no shutdown
----------------------------------------------
*A:ALA-Dut-B>config>service>epipe#
To disable the control word on spoke-sdp 1:2001:
*A:ALA-Dut-B>config>service>epipe# info
----------------------------------------------
     description "Default epipe description for service id 2100"
     sap 1/2/7:4 create
          description "Default sap description for service id 2100"
     exit
     spoke-sdp 1:2001 create
     exit
     no shutdown
----------------------------------------------
*A:ALA-Dut-B>config>service>epipe#
```

Same Fate Epipe VLANs Access Protection

The following displays a G.8031 Ethernet Tunnel for Epipe protection configuration example using same-fate SAPs for each Epipe access (two ethernet member ports 1/1/1 and 2/1/1/1 are used):

```
*A:7750_ALU>config>eth-tunnel 1
-----------------------------------------------
        description "Protection is APS"
        protection-type 8031_1to1
        ethernet
            mac 00:11:11:11:11:12
            encap-type dot1q
        exit
        ccm-hold-time down 5 up 10 // 50 ms down, 1 second up
        path 1
            member 1/1/1
            control-tag 5 // primary control vlan 5
            precedence primary
            eth-cfm
                mep 2 domain 1 association 1
                    ccm-enable
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        path 2
            member 2/1/1
            control-tag 105 //secondary control vlan 105
            eth-cfm
                mep 2 domain 1 association 2
                    ccm-enable
                    control-mep
                    no shutdown
                exit
            exit
            no shutdown
        exit
        no shutdown
-------------------------------------------------
# Configure Ethernet tunnel SAPs
-------------------------------------------------
*A:7750_ALU>config>service epipe 10 customer 5 create
        sap eth-tunnel-1 create // Uses control tags from the Ethernet tunnel port
            description "g8031-protected access ctl/data SAP for eth-tunnel 1"

        exit
        no shutdown
-----------------------------------------------
*A:7750_ALU>config>service epipe 11 customer 5 create
    sap eth-tunnel-1:1 create
        description "g8031-protected access same-fate SAP for eth-tunnel 1"

        // must specify tags for each corresponding path in Ethernet tunnel port
        eth-tunnel path 1 tag 6
        eth-tunnel path 2 tag 106
    exit
    …
```

```
-----------------------------------------------
*A:7750_ALU>config>service epipe 10 customer 5 create
    sap eth-tunnel-1:3 create
        description "g8031-protected access same-fate SAP for eth-tunnel 1"
        // must specify tags for each path for same-fate SAPs
        eth-tunnel path 1 tag 10
        eth-tunnel path 2 tag 110
    exit
    …
-----------------------------------------------
```

Pseudowire Configuration Notes

The **vc-switching** parameter must be specified at the time the VLL service is created. Note that when the **vc-switching** parameter is specified, you are configuring an S-PE. This is a pseudowire switching point (switching from one pseudowire to another). Therefore, you cannot add a SAP to the configuration.

The following example show the configuration when a SAP is added to a pseudowire. The CLI generates an error response if you attempt to create a SAP. VC switching is only needed on the pseudowire at the S-PE.

```
*A:ALA-701>config>service# epipe 28 customer 1 create vc-switching
*A:ALA-701>config>service>epipe$ sap 1/1/3 create
MINOR: SVCMGR #1311 SAP is not allowed under PW switching service
*A:ALA-701>config>service>epipe$
```

Use the following CLI syntax to create pseudowire switching VLL services.

CLI Syntax: `config>service# apipe service-id [customer customer-id] [vpn vpn-id] [vc-type {atm-vcc|atm-sdu|atm-vpc|atm-cell}] [vc-switching]`
`description description-string`
`spoke-sdp sdp-id:vc-id`

CLI Syntax: `config>service# epipe service-id [customer customer-id][vpn vpn-id][vc-switching]`
`description description-string`
`spoke-sdp sdp-id:vc-id`

CLI Syntax: `config>service# fpipe service-id [customer customer-id][vpn vpn-id] [vc-type {fr-dlci}] [vc-switching]`
`description description-string`
`spoke-sdp sdp-id:vc-id`

CLI Syntax: `config>service# ipipe service-id [customer customer-id][vpn vpn-id] [vc-switching]`
`description description-string`
`spoke-sdp sdp-id:vc-id`

The following displays an example of the command usage to configure VLL pseudowire switching services:

Example:`config>service# apipe 1 customer 1 vpn 1 vc-switching create`

```
config>service>apipe$ description "Default apipe description for
service id 100"
config>service>apipe# spoke-sdp 3:1 create
config>service>apipe>spoke-sdp# exit
config>service>apipe# spoke-sdp 6:200 create
config>service>apipe>spoke-sdp# exit
config>service>apipe# no shutdown
```

The following example displays configurations for each service:

```
*A:ALA-48>config>service# info
----------------------------------------------
...
        apipe 100 customer 1 vpn 1 vc-switching create
            description "Default apipe description for service id 100"
            spoke-sdp 3:1 create
            exit
            spoke-sdp 6:200 create
            exit
            no shutdown
        exit
...
        epipe 107 customer 1 vpn 107 vc-switching create
            description "Default epipe description for service id 107"
            spoke-sdp 3:8 create
            exit
            spoke-sdp 6:207 create
            exit
            no shutdown
        exit
...
        ipipe 108 customer 1 vpn 108 vc-switching create
            description "Default ipipe description for service id 108"
            spoke-sdp 3:9 create
            exit
            spoke-sdp 6:208 create
            exit
            no shutdown
        exit
...
        fpipe 109 customer 1 vpn 109 vc-switching create
            description "Default fpipe description for service id 109"
            spoke-sdp 3:5 create
            exit
            spoke-sdp 6:209 create
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-48>config>service#
```

Configuring Two VLL Paths Terminating on T-PE2

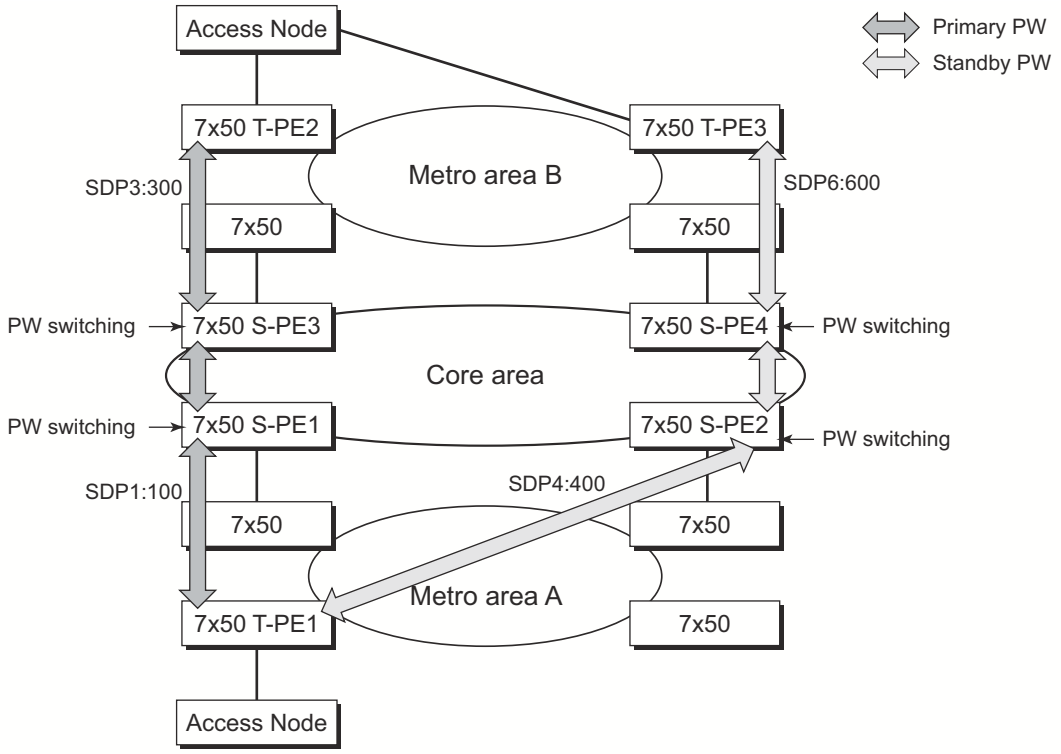

Figure 38: VLL Resilience with Pseudowire Redundancy and Switching

T-PE1

The following displays an example of the T-PE1 configuration.

```
*A:ALA-T-PE1>config>service>epipe# info
----------------------------------------------
            endpoint "x" create
            exit
            endpoint "y" create
            exit
            spoke-sdp 1:100 endpoint "y" create
                precedence primary
                revert-time 0
            exit
            spoke-sdp 4:400 endpoint "y" create
                precedence 0
            exit
            no shutdown
----------------------------------------------
*A:ALA-T-PE1>config>service>epipe#
```

The following displays an example of the T-PE2 configuration.

T-PE2

```
*A:ALA-T-PE2>config>service>epipe# info
----------------------------------------------
            endpoint "x" create
            exit
            endpoint "y" create
            exit
            sap 2/2/2:200 endpoint "x" create
            exit
            spoke-sdp 3:300 endpoint "y" create
                precedence primary
                revert-time 0
            exit
            spoke-sdp 6:600 endpoint "y" create
                precedence 0
            exit
            no shutdown
----------------------------------------------
*A:ALA-T-PE2>config>service>epipe#
```

S-PE1: Note that specifying the **vc-switching** parameter enables a VC cross-connect so the service manager does not signal the VC label mapping immediately but will put this into passive mode.

The following example displays the configuration:

```
*A:ALA-S-PE1>config>service>epipe# info
----------------------------------------------
...
            spoke-sdp 2:200 create
            exit
            spoke-sdp 3:300 create
            exit
            no shutdown
----------------------------------------------
*A:ALA-S-PE1>config>service>epipe#
```

S-PE2: Note that specifying the **vc-switching** parameter enables a VC cross-connect so the service manager does not signal the VC label mapping immediately but will put this into passive mode.

The following example displays the configuration:

```
*A:ALA-S-PE2>config>service>epipe# info
----------------------------------------------
...
            spoke-sdp 2:200 create
            exit
            spoke-sdp 3:300 create
            exit
            no shutdown
----------------------------------------------
*A:ALA-S-PE2>config>service>epipe#
```

Configuring VLL Resilience

Figure 39 displays an example to create VLL resilience. Note that the zero revert-time value means that the VLL path will be switched back to the primary immediately after it comes back up.

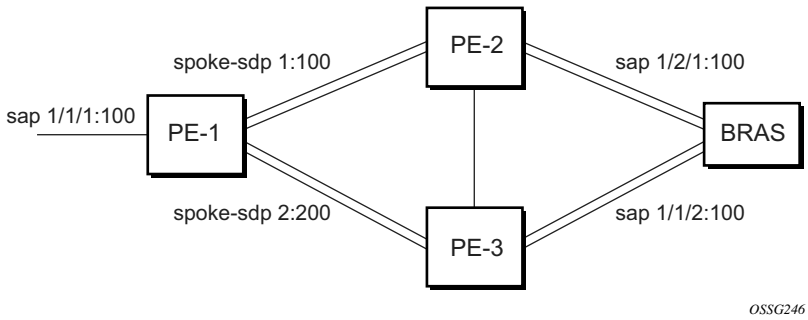

Figure 39: VLL Resilience

PE1:

The following displays an example for the configuration on PE1.

```
*A:ALA-48>config>service>epipe# info
----------------------------------------------
            endpoint "x" create
            exit
            endpoint "y" create
            exit
            spoke-sdp 1:100 endpoint "y" create
                precedence primary
            exit
            spoke-sdp 2:200 endpoint "y" create
                precedence 1
            exit
            no shutdown
----------------------------------------------
*A:ALA-48>config>service>epipe#
```

Configuring VLL Resilience for a Switched Pseudowire Path

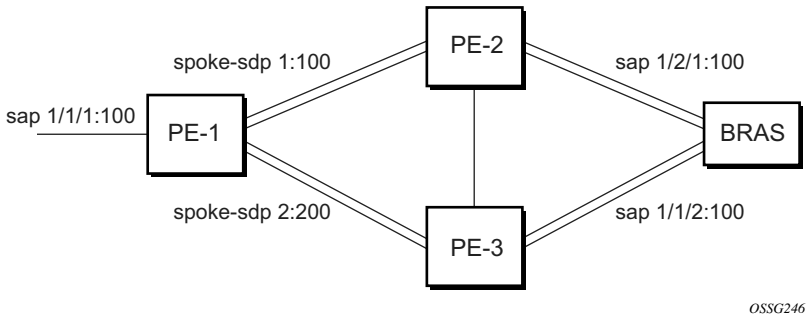

Figure 40: VLL Resilience with Pseudowire Switching

T-PE1

The following displays an example for the configuration on TPE1.

```
*A:ALA-48>config>service>epipe# info
----------------------------------------------
            endpoint "x" create
            exit
            endpoint "y" create
            exit
            sap 1/1/1:100 endpoint "x" create
            exit
            spoke-sdp 1:100 endpoint "y" create
                precedence primary
            exit
            spoke-sdp 2:200 endpoint "y" create
                precedence 1
            exit
            spoke-sdp 3:300 endpoint "y" create
                precedence 1
            exit
            no shutdown
----------------------------------------------
*A:ALA-48>config>service>epipe#
```

T-PE2

The following displays an example for the configuration on TPE2.

```
*A:ALA-49>config>service>epipe# info
----------------------------------------------
            endpoint "x" create
            exit
            endpoint "y" create
                revert-time 100
            exit
            spoke-sdp 4:400 endpoint "y" create
                precedence primary
            exit
            spoke-sdp 5:500 endpoint "y" create
                precedence 1
            exit
            spoke-sdp 6:600 endpoint "y" create
                precedence 1
            exit
            no shutdown
----------------------------------------------
*A:ALA-49>config>service>epipe#
```

S-PE1

The following displays an example for the configuration on S-PE1.

```
*A:ALA-50>config>service>epipe# info
----------------------------------------------
...
            spoke-sdp 1:100 create
            exit
            spoke-sdp 4:400 create
            exit
            no shutdown
----------------------------------------------
*A:ALA-49>config>service>epipe#
```

Configuring BGP Virtual Private Wire Service (VPWS)

Single-Homed BGP VPWS

Figure 41 shows an example topology for a BGP VPWS service used to create a virtual lease-line across an MPLS network between two sites, A and B.

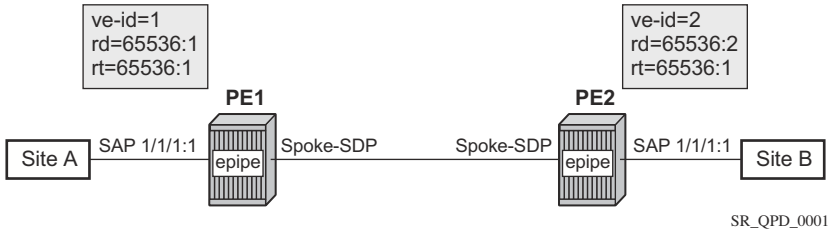

Figure 41: Single-Homed BGP VPWS Configuration Example

An Epipe is configured on PE1 and PE2 with BGP VPWS enabled. PE1 and PE2 are connected to site A and B, respectively, each using a SAP. The interconnection between the two PEs is achieved through a pseudowire, using Ethernet VLAN encaps, which is signaled using BGP VPWS over a tunnel LSP between PE1 and PE2. A MIP or MEP can be configured on a BGP VPWS SAP. However, fault propagation between a MEP and the BGP update state signaling is not supported. BGP VPWS routes are accepted only over an iBGP session.

The following displays the BGP VPWS configuration on each PE.

```
PE1:
pw-template 1 create
    vc-type vlan
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:1
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
    bgp-vpws
        ve-name PE1
            ve-id 1
        exit
        remote-ve-name PE2
            ve-id 2
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    no shutdown
exit
```

```
PE2:

pw-template 1 create
    vc-type vlan
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:2
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
    bgp-vpws
        ve-name PE2
            ve-id 2
        exit
        remote-ve-name PE1
            ve-id 1
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    no shutdown
exit
```

The BGP-VPWS update can be shown using the following command:

```
A:PE1# show service l2-route-table bgp-vpws detail
===============================================================================
Services: L2 Bgp-Vpws Route Information - Summary
===============================================================================
Svc Id          : 1
VeId            : 2
PW Temp Id      : 1
RD              : *65536:2
Next Hop        : 1.1.1.2
State (D-Bit)   : up(0)
Path MTU        : 1514
Control Word    : 0
Seq Delivery    : 0
Status          : active
Tx Status       : active
CSV             : 0
Preference      : 0
Sdp Bind Id     : 17407:4294967295
===============================================================================
A:PE1#
```

Dual-Homed BGP VPWS

Single Pseudowire Example:

Figure 42 shows an example topology for a dual-homed BGP VPWS service used to create a virtual lease-line across an MPLS network between two sites, A and B. A single pseudowire is established between the designated forwarder of the dual-homed PEs and the remote PE.

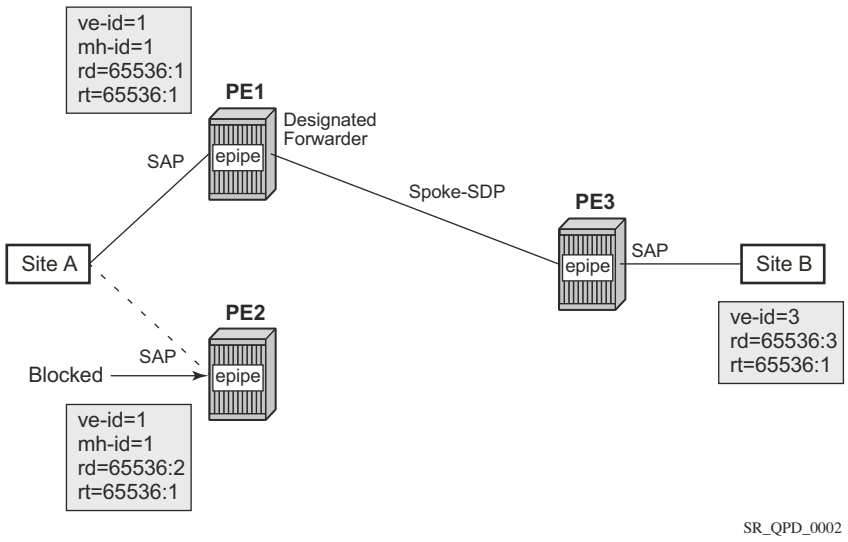

Figure 42: Example of Dual-Homed BGP VPWS with Single Pseudowire

An Epipe with BGP VPWS enabled is configured on each PE. Site A is dual-homed to PE1 and PE2 with a remote PE, PE3, connected to site B; each connection uses a SAP. A single pseudowire using Ethernet Raw Mode encaps connects PE3 to PE1. The pseudowire is signaled using BGP VPWS over a tunnel LSPs between the PEs.

Site A is configured on PE1 and PE2 with the BGP route selection, the site state, and the site-preference used to ensure PE1 is the designated forwarder when the network is fully operational.

The following displays the BGP VPWS configuration on each PE.

PE1:

```
pw-template 1 create
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:1
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
```

```
    bgp-vpws
        ve-name PE1
            ve-id 1
        exit
        remote-ve-name PE3
            ve-id 3
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    site "siteA" create
        site-id 1
        sap 1/1/1:1
        boot-timer 20
        site-activation-timer 5
        no shutdown
    exit
    no shutdown
exit
```

PE2:

```
pw-template 1 create
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:2
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
    bgp-vpws
        ve-name PE2
            ve-id 1
        exit
        remote-ve-name PE3
            ve-id 3
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    site "siteA" create
        site-id 1
        sap 1/1/1:1
        boot-timer 20
        site-activation-timer 5
        no shutdown
    exit
    no shutdown
exit
```

PE3:

```
pw-template 1 create
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:3
```

```
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
    bgp-vpws
        ve-name PE3
            ve-id 3
        exit
        remote-ve-name PE1orPE2
            ve-id 1
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    no shutdown
exit
```

Active/Standby Pseudowire Example:

Figure 43 shows an example topology for a dual-homed BGP VPWS service used to create a virtual lease-line across an MPLS network between two sites, A and B. Two pseudowires are established between the remote PE and the dual-homed PEs. The active pseudowire used for the traffic is the one connecting the remote PE to the designated forwarder of the dual-homed PEs.

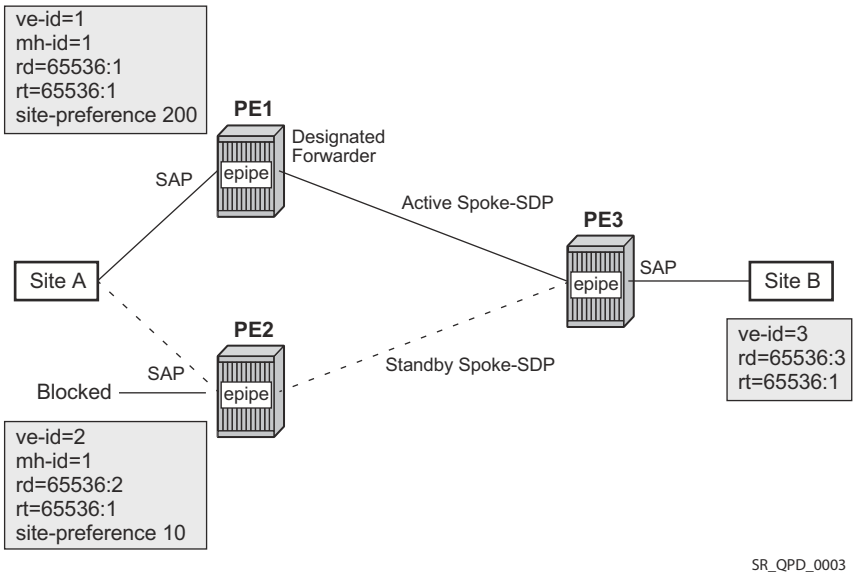

Figure 43: Example of Dual-homed BGP VPWS with Active/Standby Pseudowires

An Epipe with BGP VPWS enabled is configured on each PE. Site A is dual-homed to PE1 and PE2 with a remote PE, PE3, connected to site B; each connection uses a SAP. Active/standby pseudowires using Ethernet Raw Mode encaps connect PE3 to PE1 and PE2, respectively. The pseudowires are signaled using BGP VPWS over a tunnel LSPs between the PEs.

Site A is configured on PE1 and PE2 with the site-preference set to ensure that PE1 is the designated forwarder when the network is fully operational. An endpoint is automatically created on PE3 in which the active/standby pseudowires are created.

The following displays the BGP VPWS configuration on each PE.

PE1:

```
pw-template 1 create
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:1
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
```

```
    exit
    bgp-vpws
        ve-name PE1
            ve-id 1
        exit
        remote-ve-name PE3
            ve-id 3
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    site "siteA" create
        site-id 1
        sap 1/1/1:1
        boot-timer 20
        site-activation-timer 5
        site-preference 200
        no shutdown
    exit
    no shutdown
exit
```

PE2:

```
pw-template 1 create
exit
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:2
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
    bgp-vpws
        ve-name PE2
            ve-id 2
        exit
        remote-ve-name PE3
            ve-id 3
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    site "siteA" create
        site-id 1
        sap 1/1/1:1
        boot-timer 20
        site-activation-timer 5
        site-preference 10
        no shutdown
    exit
    no shutdown
exit
```

PE3:

```
pw-template 1 create
exit
```

```
epipe 1 customer 1 create
    bgp
        route-distinguisher 65536:3
        route-target export target:65536:1 import target:65536:1
        pw-template-binding 1
        exit
    exit
    bgp-vpws
        ve-name PE3
            ve-id 3
        exit
        remote-ve-name PE1
            ve-id 1
        exit
        remote-ve-name PE2
            ve-id 2
        exit
        no shutdown
    exit
    sap 1/1/1:1 create
    exit
    no shutdown
exit
```

Service Management Tasks

This section discusses the following Apipe service management tasks:

This section discusses the following Cpipe service management tasks:

This section discusses the following Epipe service management tasks:

This section discusses the following Fpipe service management tasks:

This section discusses the following Ipipe service management tasks:

Modifying Apipe Service Parameters

The following example displays command usage to modify Apipe parameters:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# sap 1/1/1:0/32 create
A:ALA-41>config>service>apipe>sap# accounting-policy 2
A:ALA-41>config>service>apipe>sap# exit
A:ALA-41>config>service>apipe# spoke-sdp 1:4
A:ALA-41>config>service>apipe>spoke-sdp# egress
A:ALA-41>config>service>apipe>spoke-sdp>egress# vc-label 16
A:ALA-41>config>service>apipe>spoke-sdp>egress# exit
A:ALA-41>config>service>apipe>spoke-sdp# exit
A:ALA-41>config>service>apipe#
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# apipe 5
A:ALA-42>config>service>apipe# sap 2/2/2:0/32 create
A:ALA-42>config>service>apipe>sap# accounting-policy 2
A:ALA-42>config>service>apipe>sap# exit
A:ALA-42>config>service>apipe# spoke-sdp 1:4
A:ALA-42>config>service>apipe>spoke-sdp# egress
A:ALA-42>config>service>apipe>spoke-sdp>egress# vc-label 16
A:ALA-42>config>service>apipe>spoke-sdp>egress# exit
A:ALA-42>config>service>apipe>spoke-sdp# exit
A:ALA-42>config>service>apipe#
```

```
PE Router 1 (ALA-41):
A:ALA-41>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            sap 1/1/1:0/32 create
               accounting-policy 2
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
            spoke-sdp 1:4 create
               egress
                   vc-label 16
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-41>config>service#
```

```
PE Router 2 (ALA-42):
A:ALA-42>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            description "apipe test"
            service-mtu 1400
            sap 2/2/2:0/32 create
               accounting-policy 2
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
            spoke-sdp 1:4 create
               egress
                   vc-label 16
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-42>config>service#
```

Disabling an Apipe Service

An Apipe service can be shut down without deleting any service parameters.

CLI Syntax:
```
config>service#
        apipe service-id
           shutdown
```

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# shutdown
A:ALA-41>config>service>apipe# exit
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# apipe 5
A:ALA-42>config>service>apipe# shutdown
A:ALA-42>config>service>apipe# exit
```

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            shutdown
            description "apipe test"
            service-mtu 1400
            sap 1/1/1:0/32 create
               accounting-policy 2
               ingress
                   qos 102
               exit
               egress
                   qos 103
               exit
            exit
            spoke-sdp 1:4 create
               egress
                   vc-label 16
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
------------------------------------
...
        apipe 5 customer 1 create
            shutdown
            description "apipe test"
            service-mtu 1400
            sap 2/2/2:0/32 create
               accounting-policy 2
               ingress
                   qos 102
```

```
                exit
                egress
                    qos 103
                exit
             exit
             spoke-sdp 1:4 create
                egress
                    vc-label 16
             exit
          exit
...
-----------------------------------
A:ALA-42>config>service#
```

Re-Enabling an Apipe Service

To re-enable an Apipe service that was shut down.

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

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# no shutdown
A:ALA-41>config>service>apipe# exit
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# apipe 5
A:ALA-42>config>service>apipe# no shutdown
A:ALA-42>config>service>apipe# exit
```

Deleting an Apipe Service

An Apipe service cannot be deleted until the SAP is shut down. If protocols and/or a spoke-SDP are defined, they must be shut down and removed from the configuration as well.

Use the following CLI syntax to delete Apipe services:

CLI Syntax:
```
config>service#
    no apipe service-id
       shutdown
       no sap sap-id
          shutdown
       no spoke-sdp [sdp-id:vc-id]
          shutdown
```

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# sap 1/1/1:0/32
A:ALA-41>config>service>apipe>sap# shutdown
A:ALA-41>config>service>apipe>sap# exit
A:ALA-41>config>service>apipe# no sap 1/1/1:0/32
A:ALA-41>config>service>apipe# spoke-sdp 1:4
A:ALA-41>config>service>apipe>spoke-sdp# shutdown
A:ALA-41>config>service>apipe>spoke-sdp# exit
A:ALA-41>config>service>apipe# no spoke-sdp 1:4
A:ALA-41>config>service>apipe# shutdown
A:ALA-41>config>service>apipe# exit
A:ALA-41>config>service# no apipe 5
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-41>config>service# apipe 5
A:ALA-41>config>service>apipe# sap 2/2/2:0/32
A:ALA-41>config>service>apipe>sap# shutdown
A:ALA-41>config>service>apipe>sap# exit
A:ALA-41>config>service>apipe# no sap 2/2/2:0/32
A:ALA-41>config>service>apipe# spoke-sdp 1:4
A:ALA-41>config>service>apipe>spoke-sdp# shutdown
A:ALA-41>config>service>apipe>spoke-sdp# exit
A:ALA-41>config>service>apipe# no spoke-sdp 1:4
A:ALA-41>config>service>apipe# shutdown
A:ALA-41>config>service>apipe# exit
A:ALA-41>config>service# no apipe 5
```

Modifying a Cpipe Service

The following example displays the Cpipe service configuration.

```
*A:ALA-1>config>service# info
----------------------------------------------
...
        cpipe 94002 customer 1 vc-type cesopsn create
            endpoint "to7705" create
            exit
            endpoint "toMC-APS" create
            exit
            sap aps-4.1.1.2.1 endpoint "toMC-APS" create
                ingress
                    qos 20
                exit
            exit
            spoke-sdp 14004:94002 endpoint "to7705" create
            exit
            spoke-sdp 100:294002 endpoint "toMC-APS" icb create
            exit
            spoke-sdp 100:194002 endpoint "to7705" icb create
            exit
            no shutdown
        exit
...
----------------------------------------------
*A:ALA-1>config>service> Cpipe#
```

Deleting a Cpipe Service

A Cpipe service cannot be deleted until SAPs are shut down and deleted. If a spoke-SDP is defined, it must be shut down and removed from the configuration as well.

Use the following CLI syntax to delete a Cpipe service:

CLI Syntax:
```
config>service#
        [no] cpipe service-id [customer customer-id]
           [no] spoke-sdp sdp-id
              [no] shutdown
           shutdown
```

Modifying Epipe Service Parameters

The following displays an example of adding an accounting policy to an existing SAP:

```
Example:config>service# epipe 2
        config>service>epipe# sap 2/1/3:21
        config>service>epipe>sap# accounting-policy 14
        config>service>epipe>sap# exit
```

The following output displays the SAP configuration:

```
ALA-1>config>service# info
----------------------------------------------
     epipe 2 customer 6 vpn 2 create
            description "Distributed Epipe service to east coast"
            sap 2/1/3:21 create
              accounting-policy 14
            exit
            spoke-sdp 2:6000 create
            exit
            no shutdown
        exit
----------------------------------------------
ALA-1>config>service#
```

Disabling an Epipe Service

You can shut down an Epipe service without deleting the service parameters.

CLI Syntax: `config>service> epipe` *service-id*
shutdown

Example:
```
config>service# epipe 2
        config>service>epipe# shutdown
        config>service>epipe# exit
```

Re-Enabling an Epipe Service

To re-enable an Epipe service that was shut down.

CLI Syntax:
```
config>service# epipe service-id
               no shutdown
```

```
Example:config>service# epipe 2
       config>service>epipe# no shutdown
       config>service>epipe# exit
```

Deleting an Epipe Service

Perform the following steps prior to deleting an Epipe service:

1. Shut down the SAP and SDP.
2. Delete the SAP and SDP.
3. Shut down the service.

Use the following CLI syntax to delete an Epipe service:

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

Example:
```
config>service# epipe 2
       config>service>epipe# sap 2/1/3:21
       config>service>epipe>sap# shutdown
       config>service>epipe>sap# exit
       config>service>epipe# no sap 2/1/3:21
       config>service>epipe# spoke-sdp 2:6000
       config>service>epipe>spoke-sdp# shutdown
       config>service>epipe>spoke-sdp# exit
       config>service>epipe# no spoke-sdp 2:6000
       config>service>epipe# epipe 2
       config>service>epipe# shutdown
       config>service>epipe# exit
       config>service# no epipe 2
```

Modifying Fpipe Service Parameters

The following example displays command usage to modify Fpipe parameters:

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# sap 1/2/1:16 create
A:ALA-41>config>service>fpipe>sap# accounting-policy 2
A:ALA-41>config>service>fpipe>sap# exit
A:ALA-41>config>service>fpipe# spoke-sdp 1:4
A:ALA-41>config>service>fpipe>spoke-sdp# ingress
A:ALA-41>config>service>fpipe>spoke-sdp>filter ip 10
A:ALA-41>config>service>fpipe>spoke-sdp# exit
A:ALA-41>config>service>fpipe#
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# fpipe 1
A:ALA-42>config>service>fpipe# sap 2/1/1.1:16 create
A:ALA-42>config>service>fpipe>sap# accounting-policy 2
A:ALA-42>config>service>fpipe>sap# exit
A:ALA-42>config>service>fpipe# spoke-sdp 1:1
A:ALA-42>config>service>fpipe>spoke-sdp# egress
A:ALA-42>config>service>fpipe>spoke-sdp>egress# filter ip 10
A:ALA-42>config>service>fpipe>spoke-sdp>egress# exit
A:ALA-42>config>service>fpipe>spoke-sdp# exit
A:ALA-42>config>service>fpipe#
```

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            sap 1/2/1:16 create
               accounting-policy 2
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            spoke-sdp 1:1 create
               ingress
                   filter ip 10
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
------------------------------------
...
        fpipe 1 customer 1 create
            description "fpipe test"
            service-mtu 1400
            sap 2/1/1.1:16 create
               accounting-policy 2
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            spoke-sdp 1:1 create
               egress
                   filter ip 10
            exit
            no shutdown
        exit
...
------------------------------------
A:ALA-42>config>service#
```

Disabling an Fpipe Service

An Fpipe service can be shut down without deleting any service parameters.

CLI Syntax:
```
config>service#
        fpipe service-id
           shutdown
```

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# shutdown
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# fpipe 1
A:ALA-42>config>service>fpipe# shutdown
```

PE Router 1 (ALA-41):

```
A:ALA-41>config>service# info
------------------------------------
...
        fpipe 1 customer 1 create
            shutdown
            description "fpipe test"
            service-mtu 1400
            sap 1/2/1:16 create
               accounting-policy 2
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            spoke-sdp 1:1 create
               ingress
                   filter ip 10
            exit
        exit
...
------------------------------------
A:ALA-41>config>service#
```

PE Router 2 (ALA-42):

```
A:ALA-42>config>service# info
------------------------------------
...
        fpipe 1 customer 1 create
```

```
            shutdown
            description "fpipe test"
            service-mtu 1400
            sap 2/1/1.1:16 create
               accounting-policy 2
               ingress
                   qos 101
               exit
               egress
                   qos 1020
               exit
            exit
            spoke-sdp 1:1 create
               egress
                   filter ip 10
            exit
         exit
...
------------------------------------
A:ALA-42>config>service#
```

Re-enabling an Fpipe Service

To re-enable an Fpipe service that was shut down.

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

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# no shutdown
A:ALA-41>config>service>fpipe# exit
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-42>config>service# fpipe 1
A:ALA-42>config>service>fpipe# no shutdown
A:ALA-42>config>service>fpipe# exit
```

Deleting an Fpipe Service

An Fpipe service cannot be deleted until the SAP is shut down. If protocols and/or a spoke-SDP are defined, they must be shut down and removed from the configuration as well.

Use the following CLI syntax to delete a Fpipe service:

CLI Syntax:
```
config>service#
      no fpipe service-id
         shutdown
         no sap sap-id
            shutdown
         no spoke-sdp [sdp-id:vc-id]
            shutdown
```

PE router 1 (A:ALA-41):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# sap 1/1/1:0/32
A:ALA-41>config>service>fpipe>sap# shutdown
A:ALA-41>config>service>fpipe>sap# exit
A:ALA-41>config>service>fpipe# no sap 1/1/1:0/32
A:ALA-41>config>service>fpipe# spoke-sdp 1:1
A:ALA-41>config>service>fpipe>spoke-sdp# shutdown
A:ALA-41>config>service>fpipe>spoke-sdp# exit
A:ALA-41>config>service>fpipe# no spoke-sdp 1:1
A:ALA-41>config>service>fpipe# shutdown
A:ALA-41>config>service>fpipe# exit
A:ALA-41>config>service# no fpipe 1
```

PE router 2 (A:ALA-42):

Example:
```
A:ALA-41>config>service# fpipe 1
A:ALA-41>config>service>fpipe# sap 2/1/1.1:16
A:ALA-41>config>service>fpipe>sap# shutdown
A:ALA-41>config>service>fpipe>sap# exit
A:ALA-41>config>service>fpipe# no sap 2/1/1.1:16
A:ALA-41>config>service>fpipe# spoke-sdp 1:1
A:ALA-41>config>service>fpipe>spoke-sdp# shutdown
A:ALA-41>config>service>fpipe>spoke-sdp# exit
A:ALA-41>config>service>fpipe# no spoke-sdp 1:1
A:ALA-41>config>service>fpipe# shutdown
A:ALA-41>config>service>fpipe# exit
A:ALA-41>config>service# no fpipe 1
```

Modifying Ipipe Service Parameters

The following example displays command usage to modify Ipipe parameters:

Example:
```
config>service# ipipe 202
config>service>ipipe# sap 1/1/2:444
config>service>ipipe>sap# shutdown
config>service>ipipe>sap# exit
config>service>ipipe# no sap 1/1/2:444
config>service>ipipe# sap 1/1/2:555 create
config>service>ipipe>sap$ description "eth_ipipe"
config>service>ipipe>sap$ ce-address 31.31.31.1
config>service>ipipe>sap$ no shutdown
config>service>ipipe>sap$ exit
config>service>ipipe# info
```

```
A:ALA-48>config>service# info
----------------------------------------------
...
        ipipe 202 customer 1 create
            sap 1/1/2:445 create
                description "eth_ipipe"
                ce-address 31.31.31.2
            exit
            sap 1/1/2:555 create
                description "eth_ipipe"
                ce-address 31.31.31.1
            exit
            no shutdown
        exit
...
----------------------------------------------
A:ALA-48>config>service#
```

Disabling an Ipipe Service

An Ipipe service can be shut down without deleting any service parameters.

CLI Syntax:
```
config>service#
ipipe service-id
   shutdown
```

Example:
```
A:ALA-41>config>service# ipipe 202
A:ALA-41>config>service>ipipe# shutdown
```

```
A:ALA-48>config>service# info
----------------------------------------------
...
       ipipe 202 customer 1 create
           shutdown
           sap 1/1/2:445 create
               description "eth_ipipe"
               ce-address 31.31.31.2
           exit
           sap 1/1/2:555 create
               description "eth_ipipe"
               ce-address 31.31.31.1
           exit
       exit
...
----------------------------------------------
A:ALA-48>config>service#
```

Re-enabling an Ipipe Service

To re-enable an Ipipe service that was shut down.

CLI Syntax:
```
config>service#
ipipe service-id
   no shutdown
```

Example:
```
A:ALA-41>config>service# ipipe 202
A:ALA-41>config>service>ipipe# no shutdown
```

Deleting an Ipipe Service

An Ipipe service cannot be deleted until the SAP is shut down. If protocols and/or a spoke-SDP are defined, they must be shut down and removed from the configuration as well.

Use the following CLI syntax to delete an Ipipe service:

CLI Syntax:
```
config>service#
no ipipe service-id
   shutdown
   no sap sap-id
      shutdown
   no spoke-sdp [sdp-id:vc-id]
      shutdown
```

Example:
```
config>service# ipipe 207
config>service>ipipe# sap 1/1/2:449
config>service>ipipe>sap# shutdown
config>service>ipipe>sap# exit
config>service>ipipe# no sap 1/1/2:449
config>service>ipipe# spoke-sdp 16:516
config>service>ipipe>spoke-sdp# shutdown
config>service>ipipe>spoke-sdp# exit
config>service>ipipe# no spoke-sdp 16:516
config>service>ipipe# exit
config>service# no ipipe 207
config>service#
```