Chapter 8
Configuring Circuit and Translational Cross-Connects

Circuit cross-connect (CCC) and translational cross-connect (TCC) allow you to configure transparent connections between two circuits, where a circuit can be a Frame Relay data-link connection identifier (DLCI), an Asynchronous Transfer Mode (ATM) virtual circuit (VC), a Point-to-Point Protocol (PPP) interface, a Cisco High-level Data Link Control (HDLC) interface, or a Multiprotocol Label Switching (MPLS) label-switched path (LSP).

Using CCC or TCC, packets from the source circuit are delivered to the destination circuit with, at most, the Layer 2 address being changed. No other processing, such as header checksums, time-to-live (TTL) decrementing, or protocol processing, is done.

To connect interfaces of the same type, use CCC. To connect unlike interfaces, use TCC.

CCC and TCC circuits fall into three categories: logical interfaces, which include ATM VCs and Frame Relay DLCIs; physical interfaces, which include PPP and Cisco HDLC; and paths, which include LSPs. The three circuit categories provide three types of cross-connect:

- Layer 2 switching (interface-to-interface)—Cross-connects between logical interfaces provide what is essentially Layer 2 switching.
- MPLS tunneling (interface-to-LSP)—Cross-connects between interfaces and LSPs allow you to connect two distant interface circuits by creating MPLS tunnels that use LSPs as the conduit.
- LSP stitching (LSP-to-LSP)—Cross-connects between LSPs provide a way to “stitch” together two label-switched paths, including paths that fall in two different traffic engineering database (TED) areas.

The cross-connect is bidirectional, so packets received on the first interface are transmitted out the second interface, and those received on the second interface are transmitted out the first interface.

For all CCC connections that connect interfaces, the interfaces must be of the same type; that is, ATM to ATM, Frame Relay to Frame Relay, PPP to PPP, or Cisco HDLC to Cisco HDLC.

For all TCC connections that connect interfaces, the interfaces can be of unlike types. Mainly, TCC is used for Layer 2.5 virtual private networks (VPNs), but it can also be used as a simple “unlike circuit” switch.

This chapter discusses the Layer 2 switching cross-connect configuration tasks. For information about MPLS tunneling and LSP stitching, see the *JUNOS MPLS Applications Configuration Guide*.

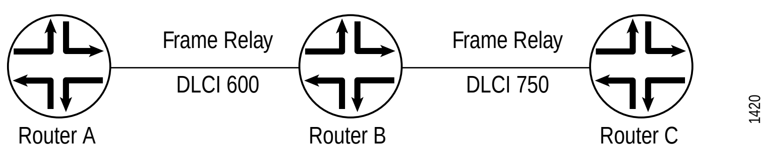
For information about Layer 2 and Layer 2.5 VPNs, see the *JUNOS VPNs Configuration Guide*.

Configuring Switching Cross-Connects

Switching cross-connects join logical interfaces to form what is essentially Layer 2 switching.

Figure 6 illustrates a Layer 2 switching circuit cross-connect. In this topology, Router A and Router C have Frame Relay connections to Router B, which is a Juniper Networks routing platform. CCC allows you to configure Router B to act as a Frame Relay (Layer 2) switch. To do this, you configure a circuit from Router A to Router C that passes through Router B, effectively configuring Router B as a Frame Relay switch with respect to these routing platforms. This configuration allows Router B to transparently switch packets (frames) between Router A and Router C without regard to the packets’ contents or the Layer 3 protocols. The only processing that Router B performs is to translate DLCI 600 to 750.

Figure 6: Layer 2 Switching Circuit Cross-Connect



If the Router A–to–Router B and Router B–to–Router C circuits are PPP, for example, the Link Control Protocol and Network Control Protocol exchanges occur between Router A and Router C. These messages are handled transparently by Router B, allowing Router A and Router C to use various PPP options (such as header or address compression and authentication) that Router B might not support. Similarly, Router A and Router C exchange keepalives, providing circuit-to-circuit connectivity status.

You can configure Layer 2 switching cross-connects on PPP, Cisco HDLC, Frame Relay, Ethernet CCC, and ATM circuits. With CCC, only like interfaces can be connected in a single cross-connect. With TCC, unlike interfaces can be connected in a single cross-connect. In Layer 2 switching cross-connects, the exchanges take place between between point-to-point links.

To configure switching cross-connects, you must configure the following on the routing platform that is acting as the switch (Router B in Figure 6 on page 140):

Defining the Encapsulation for Switching Cross-Connects

To configure Layer 2 or Layer 2.5 switching cross-connects, configure the CCC or TCC encapsulation on the routing platform that is acting as the switch (Router B in Figure 6 on page 140).

NOTE: When you use CCC encapsulation, you can configure the ccc family only. Likewise, when you use TCC encapsulation, you can configure the tcc family only.

This section is organized as follows:

Configuring PPP or Cisco HDLC Circuits

For PPP or Cisco HDLC circuits, specify the encapsulation by including the encapsulation statement at the [edit interfaces *interface-name*] hierarchy level. This statement configures the entire physical device. For these circuits to work, you must configure a logical interface unit 0.

```
[edit interfaces interface-name]
encapsulation (ppp-ccc | cisco-hdlc-ccc | ppp-tcc | cisco-hdlc-tcc);
unit 0;
```

Configuring ATM Circuits

For ATM circuits, include the vpi statement [edit interfaces *interface-name* atm-options] hierarchy level:

```
[edit interfaces at-fpc/pic/port]
atm-options {
    vpi vpi-identifier;
}
```

On the logical interface, include the following statements:

```
point-to-point;
encapsulation (atm-ccc-cell-relay | atm-ccc-vc-mux | atm-tcc-vc-mux |
        atm-tcc-snap);
vci vpi-identifier.vci-identifier;
```

You can include the logical interface statements at the following hierarchy levels:

[edit interfaces at-*fpc*/*pic*/*port* unit *logical-unit-number*]

[edit logical-routers *logical-router-name* interfaces at-*fpc*/*pic*/*port* unit *logical-unit-number*]

For each VC, you configure whether it is a circuit or a regular logical interface. The default interface type is point-to-point.

Configuring Frame Relay Circuits

For Frame Relay circuits, include the encapsulation statement at the [edit interfaces *interface-name*] hierarchy level:

```
[edit interfaces interface-name]
encapsulation type;
```

On the logical interface, include the following statements:

```
point-to-point;
encapsulation type;
dlci dlci-identifier;
```

You can include the logical interface statements at the following hierarchy levels:

[edit interfaces *interface-name* unit *logical-unit-number*]

[edit logical-routers *logical-router-name* interfaces *interface-name* unit *logical-unit-number*]

The encapsulation type can be one of the following:

- Flexible Frame Relay (flexible-frame-relay)—Intelligent queuing (IQ) interfaces can use flexible Frame Relay encapsulation. You use flexible Frame Relay encapsulation when you want to configure multiple per-unit Frame Relay encapsulations. This encapsulation type allows you to configure any combination of TCC, CCC, and standard Frame Relay encapsulations on a single physical port. Also, each logical interface can have any DLCI value from 1 through 1022.

- Frame Relay CCC version (frame-relay-ccc)—For E1, E3, SONET/SDH, T1, and T3 interfaces, this encapsulation type is the same as standard Frame Relay for DLCIs 0 through 511. DLCIs 512 through 1022 are dedicated to CCC. This numbering restriction does not apply to IQ interfaces. The logical interface must also have frame-relay-ccc encapsulation. When you use this encapsulation type, you can configure the ccc family only.

- Frame Relay TCC version (frame-relay-tcc)—Similar to Frame Relay CCC and has the same configuration restrictions, but used for circuits with different media on either side of the connection.

- Extended CCC version (extended-frame-relay-ccc)—This encapsulation type allows you to dedicate DLCIs 1 through 1022 to CCC. The logical interface must have frame-relay-ccc encapsulation. When you use this encapsulation type, you can configure the ccc family only.

- Extended TCC version (extended-frame-relay-tcc)—Similar to extended Frame Relay CCC, this encapsulation type allows you to dedicate DLCIs 1 through 1022 to TCC, which is used for circuits with different media on either side of the connection.

- Port CCC version (frame-relay-port-ccc)—Defined in the IETF document *Frame Relay Encapsulation o ver Pseudo-Wir es* (expired December 2002). This encapsulation type allows you to transparently carry all the DLCIs between two customer edge (CE) routers without explicitly configuring each DLCI on the two provider edge (PE) routers with Frame Relay transport. The connection between the two CE routers can be either user-to-network interface (UNI) or network-to-network interface (NNI); this is completely transparent to the PE routers. The logical interface does not require an encapsulation statement. When you use this encapsulation type, you can configure the ccc family only.

For each DLCI, you configure whether it is a circuit or a regular logical interface. The DLCI for regular interfaces must be from 1 through 511. For CCC and TCC interfaces, it must be from 512 through 1022. This restriction does not apply to IQ interfaces. The default interface type is point-to-point.

Configuring Ethernet CCC Circuits

You can configure Ethernet CCC encapsulation on Fast Ethernet, Gigabit Ethernet, and aggregated Ethernet interfaces.

NOTE: CCC over aggregated Ethernet requires an enhanced Flexible PIC Concentrator (FPC).

For example configurations of CCC over aggregated Ethernet, see “Examples: Configuring Switching Cross-Connects” on page 146.

For Ethernet CCC circuits, specify the encapsulation by including the encapsulation statement at the [edit interfaces *interface-name*] hierarchy level. This statement configures the entire physical device.

```
[edit interfaces interface-name]
encapsulation ethernet-ccc;
unit logical-unit-number {
…
}

[edit interfaces aex]
encapsulation ethernet-ccc;
unit logical-unit-number {
…
}
```

Configuring Ethernet VLAN Circuits

You can configure Ethernet virtual local area network (VLAN) circuits on Fast Ethernet, Gigabit Ethernet, and aggregated Ethernet interfaces. For Ethernet VLAN circuits, specify the encapsulation by including the encapsulation statement at the [edit interfaces *interface-name*] hierarchy level. This statement configures the entire physical device. You must also enable VLAN tagging. To do this, include the following statements:

```
[edit interfaces interface-name]
vlan-tagging;
encapsulation (extended-vlan-ccc | vlan-ccc);

[edit interfaces aex]
vlan-tagging;
encapsulation vlan-ccc;
```

On the logical interface, include the following statements:

```
encapsulation vlan-ccc;
vlan-id number;
```

You can include the logical interface statements at the following hierarchy levels:

[edit interfaces *interface-name* unit *logical-unit-number*]

[edit logical-routers *logical-router-name* interfaces *interface-name* unit *logical-unit-number*]

Ethernet interfaces in VLAN mode can have multiple logical interfaces. For encapsulation type vlan-ccc, VLAN IDs 1 through 511 are reserved for normal VLANs, and VLAN IDs 512 through 1023 are reserved for CCC VLANs. For encapsulation type extended-vlan-ccc, VLAN IDs 1 through 4094 are valid. VLAN ID 0 is reserved for tagging the priority of frames.

Defining the Connection for Switching Cross-Connects

To configure Layer 2 switching cross-connects, define the connection between the two circuits. You configure this on the routing platform that is acting as the switch (Router B in Figure 6 on page 140). The connection joins the interface that comes from the circuit's source to the interface that leads to the circuit's destination. When you specify the interface names, include the logical portion of the name, which corresponds to the logical unit number. The cross-connect is bidirectional, so packets received on the first interface are transmitted out the second interface, and those received on the second interface are transmitted out the first interface.

```
[edit protocols]
connections {
    remote-interface-switch connection-name {
        interface interface-name.unit-number;
        interface interface-name.unit-number;
    }
    lsp-switch connection-name {
        transmit-lsp lsp-number;
        receive-lsp lsp-number;
    }
}
```

Configuring MPLS for Switching Cross-Connects

For Layer 2 switching cross-connects to work, you must configure MPLS. The following is a minimal MPLS configuration:

```
[edit protocols]
mpls {
    interface (interface-name | all);
}
```

For more information, see the *JUNOS MPLS Applications Configuration Guide*.

Examples: Configuring Switching Cross-Connects

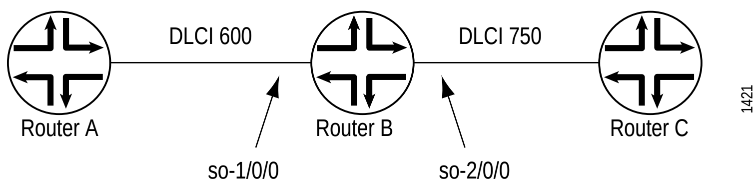
This section includes the following examples:

- Configuring a CCC over Frame Relay Encapsulated Interfaces on page 146
- Configuring a TCC on page 147
- Configuring CCC over Aggregated Ethernet on page 149
- Configuring a Remote LSP CCC over Aggregated Ethernet on page 151

Configuring a CCC over Frame Relay Encapsulated Interfaces

Configure a full-duplex Layer 2 switching circuit cross-connect between Router A and Router C, using a Juniper Networks routing platform, Router B, as the virtual switch. See the topology in Figure 7.

Figure 7: Example Topology of a Switching Circuit Cross-Connect with Frame Relay CCC Encapsulation




```
[edit]
interfaces {
    so-1/0/0 {
        encapsulation frame-relay-ccc;
        unit 1 {
            point-to-point;
            eui-64 frame-relay-ccc;
            dlci 600;
        }
    }
    so-2/0/0 {
        encapsulation frame-relay-ccc;
        unit 2 {
            point-to-point;
            encapsulation frame-relay-ccc;
            dlci 750;
        }
    }
}
```

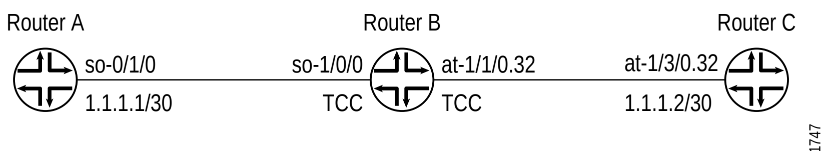
```
protocols {
    connections {
        interface-switch router-a-router-c {
            interface so-1/0/0.1;
            interface so-2/0/0.2;
        }
    }
    mpls {
        interface all;
    }
}
```

Configuring a TCC

Configure a full-duplex switching translational cross-connect with PPP TCC encapsulation between Router A and Router C, using a Juniper Networks routing platform, Router B, as the virtual switch. See the topology in Figure 8.

In this topology, Router B has a PPP connection to Router A and an ATM connection to Router C.

Figure 8: Layer 2.5 Switching Translational Cross-Connect



On Router A

```
[edit]
interfaces {
    so-0/1/0 {
        description "to Router B so-1/0/0";
        encapsulation ppp;
        unit 0 {
            family inet {
                address 10.0.0.0/32;
            }
        }
    }
}
```

On Router B

```
[edit]
interfaces {
    so-1/0/0 {
        description "to Router A so-0/1/0";
        encapsulation ppp-tcc;
        unit 0 {
        }
    }
    at-1/1/0 {
        description "to Router C at-0/3/0";
        atm-options {
            vpi 0 maximum-vcs 2000;
        }
        unit 32 {
            vci 32;
            encapsulation atm-tcc-vc-mux;
        }
    }
}
[edit]
protocols {
    mpls {
        interface so-1/0/0.0;
        interface at-1/1/0.32;
    }
connections {
    interface-switch PPP-to-ATM {
        interface so-1/0/0.0;
        interface at-1/1/0.32;
    }
}
```

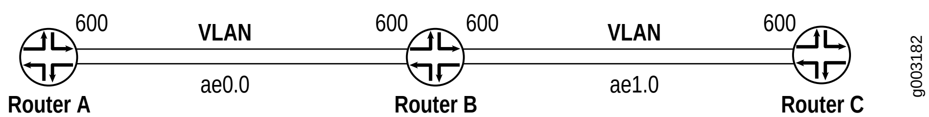
On Router C

```
[edit]
interfaces {
    at-0/3/0 {
        description "to Router B at-1/1/0";
        atm-options {
            vpi 0 maximum-vcs 2000;
        }
        unit 32 {
            vci 32;
            encapsulation atm-vc-mux;
            family inet {
                address 10.1.1.2/30;
            }
        }
    }
}
```

Configuring CCC over Aggregated Ethernet

See the topology in Figure 9. In this topology, CE Routers A and C have aggregated Ethernet connections to PE Router B. With CCC, you specify that the circuit from Router A is connected to the circuit from Router C. Router B functions as a cross-connect switch between the two circuits. For a back-to-back connection, all VLAN IDs must be the same on Router A through Router C. You configure Router A and Router C as standard aggregated Ethernet interfaces. For more information about aggregated Ethernet, see “Configuring Aggregated Ethernet Interfaces” on page 392.

Figure 9: Interface-to-Interface Circuit Cross-Connect over Aggregated Ethernet Interfaces



On Router A

```
[edit interfaces]
ae0 {
    vlan-tagging;
    aggregated-ether-options {
        minimum-links 1;
        link-speed 1g;
    }
    unit 0 {
        vlan-id 600;
        family inet {
            address 192.168.1.1/30;
        }
    }
}
```

On Router B

```
[edit interfaces]
ae0 {
    encapsulation vlan-ccc;
    vlan-tagging;
    aggregated-ether-options {
        minimum-links 1;
        link-speed 1g;
    }
    unit 0 { # CCC switch
        encapsulation vlan-ccc;
        vlan-id 600;
        family ccc;
    }
    unit 1 { # Encapsulation defaults to non-ccc
        vlan-id 600;
        family inet {
            address 10.1.1.1/24;
        }
    }
}
```

```
ae1 {
    encapsulation vlan-ccc;
    vlan-tagging;
    aggregated-ether-options {
        minimum-links 1;
        link-speed 100m;
    }
    unit 0 {
        encapsulation vlan-ccc;
        vlan-id 600;
        family ccc;
    }
}

[edit protocols]
mpls {
    interface all;
}
connections {
    interface-switch layer2-cross-connect {
        interface ae0.0;
        interface ae1.0;
    }
}
```

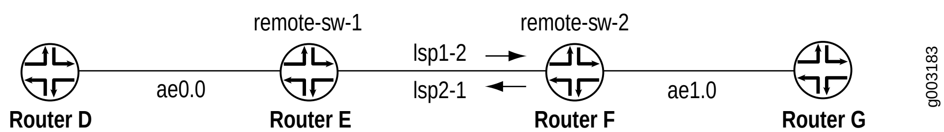
On Router C

```
[edit interfaces]
ae1 {
    vlan-tagging;
    aggregated-ether-options {
        minimum-links 1;
        link-speed 1g;
    }
    unit 0 {
        vlan-id 600;
        family inet {
            address 192.168.1.2/30;
        }
    }
}
```

Configuring a Remote LSP CCC over Aggregated Ethernet

See the topology in Figure 10. In this topology, CE Router G has an aggregated Ethernet connection to PE Router F. CE Router D has an aggregated Ethernet connection to PE Router E. Router E and Router F have an MPLS LSP between them. With remote CCC, you specify that the circuit from Router D is connected to the circuit from Router G. The circuit from Router D is connected to the LSP on Router E; the circuit from Router G is connected to the LSP on Router F. In other words, ae0.0 and ae1.0 are connected using lsp1-2 and lsp2-1. You configure Router D and Router G as standard aggregated Ethernet interfaces. For more information about aggregated Ethernet, see "Configuring Aggregated Ethernet Interfaces" on page 392.

Figure 10: Remote Interface-LSP-Interface Circuit Cross-Connect over Aggregated Ethernet Interfaces



On Router D

```
[edit interface]
ae0 {
    aggregated-ether-options {
        minimum-links 1;
        link-speed 1g;
        lacp {
            active;
            periodic fast;
        }
    }
    unit 0 {
        family inet {
            address 192.168.2.1/30;
        }
    }
}
```

On Router E

```
[edit interfaces]
ae0 {
    encapsulation ethernet-ccc;
    aggregated-ether-options {
        minimum-links 1;
        link-speed 100m;
        lacp {
            active;
            periodic fast;
        }
    }
    unit 0 {
        encapsulation vlan-ccc;   # default
        family ccc;               # default
    }
}
```

```
[edit protocols]
mpls {
    interface all;
}
connections {
    remote-interface-switch remote-sw-1 {
        interface ae1.0;
        receive-lsp lsp2_1;
        transmit-lsp lsp1_2;
    }
}
```

On Router F

```
[edit interfaces]
ae1 {
    encapsulation ethernet-ccc;
    aggregated-ether-options {
        minimum-links 1;
        link-speed 100m;
        lacp {
            active;
            periodic fast;
        }
    }
    unit 0 {
        encapsulation vlan-ccc;   # default
        family ccc;               # default
    }
}

[edit protocols]
mpls {
    interface all;
}
connections {
    remote-interface-switch remote-sw-2 {
        interface ae0.0;
        receive-lsp lsp1_2;
        transmit-lsp lsp2_1;
    }
}
```

On Router G

```
[edit interface]
ae1 {
    aggregated-ether-options {
        minimum-links 1;
        link-speed 1g;
        lacp {
            active;
            periodic fast;
        }
    }
    unit 0 {
        family inet {
            address 192.168.2.2/30;
        }
    }
```