



# CHAPTER 6

## Configuring Bridges

---

This chapter describes how to configure bridging for ML1000-2 Gigabit Ethernet cards, ML100T-12 Fast Ethernet cards, and ML100X-8 Fast Ethernet cards. For more information about the Cisco IOS commands used in this chapter, refer to the *Cisco IOS Command Reference* publication.

This chapter includes the following major sections:

- [Understanding Basic Bridging, page 6-1](#)
- [Configuring Basic Bridging, page 6-2](#)
- [Monitoring and Verifying Basic Bridging, page 6-4](#)
- [Transparent Bridging Modes of Operation, page 6-5](#)



### Caution

---

Cisco Inter-Switch Link (ISL) and Cisco Dynamic Trunking Protocol (DTP) are not supported by ML1000-2, ML100T-12, and ML100X-8 cards, but their broadcast forwards these formats. Using ISL or DTP on connecting devices is not recommended. Some Cisco devices attempt to use ISL or DTP by default.

---

## Understanding Basic Bridging

ML1000-2, ML100T-12, and ML100X-8 cards support transparent bridging for Fast Ethernet, Gigabit Ethernet and POS ports. They support a maximum of 255 active bridge groups. For information on the modes of transparent bridging, see the [“Transparent Bridging Modes of Operation” section on page 6-5](#).

To configure bridging, you must perform the following tasks in the modes indicated:

- In global configuration mode:
  - Enable bridging of IP packets.
  - Select the type of Spanning Tree Protocol (STP) (optional).
- In interface configuration mode:
  - Determine which interfaces belong to the same bridge group.

ML1000-2, ML100T-12, or ML100X-8 cards bridge all nonrouted traffic among the network interfaces comprising the bridge group. If spanning tree is enabled, the interfaces became part of the same spanning tree. Interfaces not participating in a bridge group cannot forward bridged traffic.

If the destination address of the packet is known in the bridge table, the packet is forwarded on a single interface in the bridge group. If the packet's destination is unknown in the bridge table, the packet is flooded on all forwarding interfaces in the bridge group. The bridge places source addresses in the bridge table as it learns them during the process of bridging.

Spanning tree is not mandatory for an ML1000-2, ML100T-12, or ML100X-8 bridge group. But if it is configured, a separate spanning-tree process runs for each configured bridge group. A bridge group establishes a spanning tree based on the bridge protocol data units (BPDUs) it receives on only its member interfaces.

## Configuring Basic Bridging

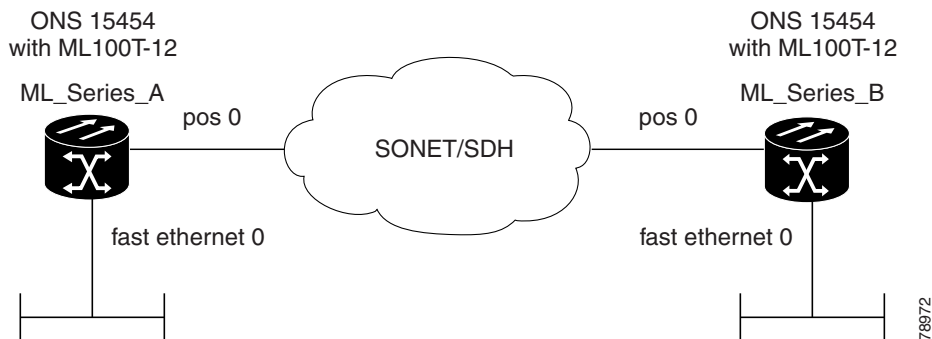
Use the following steps to configure bridging:

	Command	Purpose
Step 1	Router(config)# <b>no ip routing</b>	Enables bridging of IP packets. This command needs to be executed once per card, not once per bridge-group. This step is not done for integrated routing and bridging (IRB).
Step 2	Router(config)# <b>bridge</b> <i>bridge-group-number</i> [ <b>protocol</b> { <b>drpri-rstp</b>   <b>rstp</b>   <b>ieee</b> }]	Assigns a bridge group number and defines the appropriate spanning-tree type:  bridge-group-number can range from 1 to 4096. <ul style="list-style-type: none"> <li>• <b>drpri-rstp</b> is the protocol used to interconnect dual RPR interconnect to protect from node failure</li> <li>• <b>rstp</b> is the IEEE 802.1W Rapid Spanning Tree.</li> <li>• <b>ieee</b> is the IEEE 802.1D Spanning Tree Protocol.</li> </ul> <b>Note</b> Spanning tree is not mandatory for an ML1000-2, ML100T-12, or ML100X-8 bridge group. But configuring spanning tree blocks network loops.
Step 3	Router(config)# <b>bridge</b> <i>bridge-group-number</i> <b>priority</b> <i>number</i>	(Optional) Assigns a specific priority to the bridge, to assist in the spanning-tree root definition. Lowering the priority of a bridge makes it more likely the bridge is selected as the root.
Step 4	Router(config)# <b>interface</b> <i>type</i> <i>number</i>	Enters interface configuration mode to configure the interface of the ML1000-2, ML100T-12, or ML100X-8 card.
Step 5	Router(config-if)# <b>bridge-group</b> <i>bridge-group-number</i>	Assigns a network interface to a bridge group.
Step 6	Router(config-if)# <b>no shutdown</b>	Changes the shutdown state to up and enables the interface.
Step 7	Router(config-if)# <b>end</b>	Returns to privileged EXEC mode.
Step 8	Router# <b>copy running-config</b> <b>startup-config</b>	(Optional) Saves your entries in the configuration file.

# Bridging Examples

The ML1000-2, ML100T-12, and ML100X-8 cards all have bridging capability. In the following figures, an ML100T-12 configuration is shown as a representative model for all three cards. [Figure 6-1](#) shows a basic bridging example. [Example 6-1](#) shows the configuration of the east M100T-12 card. [Example 6-2](#) shows the configuration of the west ML100T-12.

**Figure 6-1** Bridging Example



**Example 6-1** East Router Configuration

```
bridge 1 protocol ieee
!
!
interface FastEthernet0
 no ip address
 bridge-group 1
!
interface POS0
 no ip address
 crc 32
 bridge-group 1
 pos flag c2 1
```

**Example 6-2** West Router Configuration

```
bridge 1 protocol ieee
!
!
interface FastEthernet0
 no ip address
 bridge-group 1
!
interface POS0
 no ip address
 crc 32
 bridge-group 1
 pos flag c2 1
```

# Monitoring and Verifying Basic Bridging

After you have set up an ML1000-2, ML100T-12, or ML100X-8 card for bridging, you can monitor and verify its operation by performing the following procedure in privileged EXEC mode:

	Command	Purpose
Step 1	Router# <b>clear bridge</b> <i>bridge-group-number</i>	Removes any learned entries from the forwarding database of a particular bridge group, clears the transmit, and receives counts for any statically configured forwarding entries.
Step 2	Router# <b>show bridge</b> { <i>bridge-group-number</i>   <i>interface-address</i> }	Displays classes of entries in the bridge forwarding database.
Step 3	Router# <b>show bridge verbose</b>	Displays detailed information about configured bridge groups.
Step 4	ML_Series# <b>show spanning-tree</b> { <i>bridge-group-number</i> } [ <b>brief</b> ]	Displays detailed information about spanning tree. <b>bridge-group-number</b> restricts the spanning tree information to specific bridge groups. <b>brief</b> displays summary information about spanning tree.

Example 6-3 shows an example of monitoring and verifying bridging.

## Example 6-3 Monitoring and Verifying Bridging

```

ML-Series# show bridge

Total of 300 station blocks, 298 free
Codes: P - permanent, S - self

Bridge Group 1:

Maximum dynamic entries allowed: 1000
Current dynamic entry count: 2

      Address      Action  Interface
0000.0001.6000  forward FastEthernet0
0000.0001.6100  forward POS0

ML-Series# show bridge verbose

Total of 300 station blocks, 298 free
Codes: P - permanent, S - self

Maximum dynamic entries allowed: 1000
Current dynamic entry count: 2

BG Hash      Address      Action  Interface      VC   Age   RX count  TX co
unt
  1 60/0    0000.0001.6000 forward FastEthernet0   -
  1 61/0    0000.0001.6100 forward POS0      -

Flood ports
FastEthernet0
POS0

ML-Series# show spanning-tree brief

```

```

Bridge group 1
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address    0005.9a39.6634
             This bridge is the root
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
             Address    0005.9a39.6634
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface          Role Sts Cost          Prio.Nbr Type
-----
Fa0                Desg FWD 19           128.3   P2p
PO0                Desg FWD 9            128.20  P2p

```

## Transparent Bridging Modes of Operation

The transparent bridging feature in the Cisco IOS software combines bridge-groups and IP routing. This combination provides the speed of an adaptive spanning-tree bridge, along with the functionality, reliability, and security of a router. ML1000-2, ML100T-12, and ML100X-8 cards support transparent bridging in the same general manner as other Cisco IOS platforms.

Transparent bridging processes IP frames in four distinct modes, each with different rules and configuration options. The modes are IP routing, no IP routing, bridge crb, and bridge irb. This section covers the configuration and operation of these four modes on ML1000-2, ML100T-12, and ML100X-8 cards.

For additional general Cisco IOS user documentation on configuring transparent bridging, see the “Configuring Transparent Bridging” chapter of the *Cisco IOS Bridging and IBM Networking Configuration Guide, Release 12.2* at:

[http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products\\_configuration\\_guide\\_chapter09186a00800ca767.html](http://www.cisco.com/en/US/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800ca767.html)

## IP Routing Mode

IP routing mode is the default mode. It disables the other modes (no IP routing, bridge crb, and bridge irb). The global command **ip routing** enables IP routing mode.

In IP routing mode, the bridge-groups do not process IP packets. The IP packets are either routed or discarded.

The following rules help describe packet handling in this mode:

- An input interface or subinterface configured with only a bridge-group will bridge non-IP packets and discard IP packets (Example 6-4).
- An input interface or subinterface configured with only an IP address will route IP packets and discard non-IP packets (Example 6-5).
- An input interface or subinterface configured with both an IP address and a bridge-group routes IP packets and bridges non-IP packets (Example 6-6). This configuration is sometimes referred to as fallback bridging. If a protocol cannot be routed, then the interface falls back to bridging.

- All of the interfaces or subinterfaces belonging to a specific bridge-group need consistent configuration with regards to configuring or not configuring IP addresses. Mixing interfaces configured with IP addresses and interfaces not configured with IP addresses in the same bridge group can cause inconsistent or unpredictable routing at the network level.
- All the interfaces and subinterface belonging to the same bridge-group need consistent configuration with regard to IP addresses. Either all of the bridge group's interfaces should be configured with IP addresses or none of the bridge group's interfaces should be configured with IP addresses.

[Example 6-4](#) shows card interfaces configured in a bridge group with no IP addresses.

**Example 6-4 Bridge Group with No IP Address**

```
ip routing
bridge 1 proto rstp

int f0
bridge-group 1

int pos 0
bridge-group 1
```

[Example 6-5](#) shows card interfaces configured with IP addresses but not in a bridge group.

**Example 6-5 IP Addresses with No Bridge Group**

```
ip routing

int f0
ip address 10.10.10.2 255.255.255.0

int pos 0
ip address 20.20.20.2 255.255.255.0
```

[Example 6-6](#) shows card interfaces configured with IP addresses and in a bridge group.

**Example 6-6 IP Addresses with Bridge Group**

```
ip routing
bridge 1 proto rstp

int f0
ip address 10.10.10.2 255.255.255.0
bridge-group 1

int pos 0
ip address 20.20.20.2 255.255.255.0
bridge-group 1
```

## No IP Routing Mode

The no IP routing mode bridges all packets, both IP and non-IP, and prevents routing. Although Cisco IOS can use the IP addresses for interfaces configured as management ports, it will not route between these IP addresses.

The global command **no ip routing** enables this feature, and enabling no ip routing disables the other modes.

The following rules help describe packet handling in this mode:

- An input interface or subinterface configured with only a bridge-group and no IP addresses bridges all packets (Example 6-7).
- An input interface or subinterface configured with only an IP address discards all packets, except packets with the destination MAC and IP address of the input interface, which are processed by Cisco IOS. This is not a valid configuration.
- An input interface or subinterface configured with both an IP address and a bridge group bridges all packets, except packets sent to the input interface MAC address. Packets sent to the input interface MAC address and the interface IP address are processed by Cisco IOS. Other packets sent to the input interface MAC address are discarded. This is not a valid configuration for the IP addresses.
- All of the interfaces or subinterfaces belonging to a specific bridge-group need consistent configuration in regards to configuring or not configuring IP addresses. Mixing interfaces configured with IP addresses and interfaces not configured with IP addresses in the same bridge group can cause inconsistent or unpredictable routing at the network level.

Example 6-7 shows card interfaces configured in a bridge group with no IP addresses.

**Example 6-7 Bridge Group with No IP Address**

```
no ip routing
bridge 1 proto rstp

int f0
bridge-group 1

int pos 0
bridge-group 1
```

## Bridge CRB Mode

In bridge crb mode, the default sub-mode for every bridge group is to bridge but not route the IP packets. This is similar to the no ip routing mode behavior. But with bridge crb, packet handling is configured not globally but for the specific bridge group. You can selectively disable bridge groups to block IP packets or configure fallback bridging for a group of routed interfaces.

Concurrent routing and bridging is enabled with the global command **bridge crb**. Enabling bridge crb disables the other modes.

The following rules help describe packet handling in this mode:

- The command **bridge x bridge ip** (where *x* is a bridge-group number) configures a bridge-group to bridge IP packets. Input interfaces and sub-interfaces belonging to the bridge-group will follow the rules for no IP routing mode.
- The command **bridge x route IP** (where *x* is a bridge-group number) configures a bridge-group to ignore IP packets. Input interfaces and sub-interfaces belonging to this bridge-group will follow the rules for IP routing mode (Example 6-8).
- When you enable bridge crb with pre-existing bridge groups, it will generate a **bridge x route IP** configuration command for any pre-existing bridge groups with an interface configured for routing (configured with an IP address). This is a precaution when crb is first enabled.

- All of the interfaces or subinterfaces belonging to a specific bridge-group need consistent configuration in regards to configuring or not configuring IP addresses. Mixing interfaces configured with IP addresses and interfaces not configured with IP addresses in the same bridge group can cause inconsistent or unpredictable routing at the network level.
- Routing between interfaces or subinterfaces that do not belong to the same bridge group could result in inconsistent network behavior. This mode is for routing between members of a bridge-group, but never for routing into or out of a bridge group.

Example 6-8 shows card interfaces configured with IP addresses and multiple bridge groups.

### Example 6-8 IP Addresses and Multiple Bridge Group

```
bridge crb
bridge 1 proto rstp
bridge 1 route ip
bridge 2 proto rstp

int f0
ip address 10.10.10.2 255.255.255.0
bridge-group 1

int pos 0
ip address 20.20.20.2 255.255.255.0
bridge-group 1

int f1
bridge-group 2

int pos 1
bridge-group 2
```



Tip

---

When troubleshooting a bridge crb configuration, make sure the interfaces are not assigned IP addresses belonging to the same subnet. Routing requires IP addresses to be in different subnets.

---

## Bridge IRB Mode

The integrated routing and bridging mode is enabled with the global command **bridge irb**. Enabling bridge irb disables the other modes.

Bridge irb mode is a super-set of the bridge crb mode. Only IRB mode supports a bridged virtual interface (BVI), which is a virtual Layer 3 interface belonging to a specific bridge-group. A BVI requires an IP address to function and is visible to all member interfaces of that bridge-group. The only proper way to route into and out of a bridge-group is with a BVI.

Bridge irb behaves like bridge crb with the following additions:

- If a BVI interface is configured for a bridge-group, the BVI IP address should be the only one configured on any member of that bridge-group (Example 6-9).
- If both an IP address and a bridge-group are configured on a single interface, enable either IP bridging or IP routing, but not both (Example 6-10).
- If IP routing is disabled in a bridge-group, all packets will be bridged, and BVI interfaces will not route IP. This is the default for each bridge-group.



- If IP bridging and IP routing are both enabled in a bridge-group with a BVI, then IP packets can be bridged between bridge-group members (bridging within the same subnet), and they can be routed in and out of the bridge-group via the BVI.
- If IP bridging is disabled, but IP routing is enabled in a bridge-group, IP packets can be routed in and out of the bridge-group through the BVI but cannot be bridged between the Layer 2 interfaces. The global command **bridge x route ip** in combination with the global command **no bridge x bridge ip** disables IP bridging while enabling IP routing.

**Example 6-9** shows card interfaces configured in a bridge group and the BVI configured with an IP address. Both bridging and routing are enabled.

**Example 6-9 Bridge irb with Routing and Bridging Enabled**

```
bridge irb
bridge 1 proto rstp
bridge 1 route ip

int f0
bridge-group 1

int pos 0
bridge-group 1

int bvi 1
ip address 10.10.10.1 255.255.255.0
```

**Example 6-10** shows card interfaces configured with both an IP address and a bridge-group. IP routing is enabled and IP bridging is disabled.

**Example 6-10 IP Addresses and Multiple Bridge Group**

```
bridge irb
bridge 1 proto rstp
bridge 1 route ip
no bridge 1 bridge ip

int f0
ip address 10.10.10.1 255.255.255.0
bridge-group 1

int pos 0
ip address 20.20.20.2 255.255.255.0
bridge-group 2
```



**Tip**

---

When troubleshooting bridge irb, make sure the BVI is configured with an IP address and the BVI bridge members are not configured with IP addresses.

---

