

LAN Topologies

The application in use, such as multimedia, database updates, e-mail, or file and print sharing, generally determines the type of data transmission.

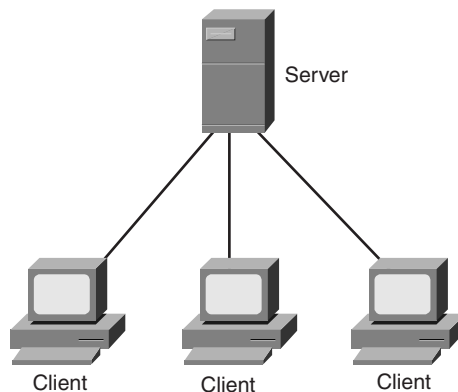
LAN transmissions fit into one of three categories:

- Unicast
- Multicast
- Broadcast

Unicast

With unicast transmissions, a single packet is sent from the source to a destination on a network. The source-node addresses the packet by using the network address of the destination node. The packet is then forwarded to the destination network and the network passes the packet to its final destination. Figure 2-1 is an example of a unicast network.

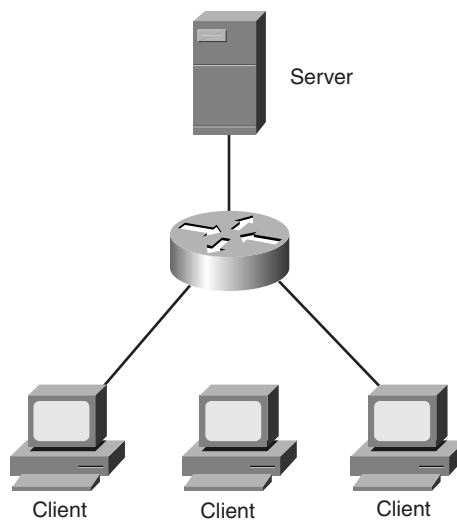
Figure 2-1 *Unicast Network*



Multicast

With a multicast transmission, a single data packet is copied and forwarded to a specific subset of nodes on the network. The source node addresses the packet by using a multicast address. For example, the TCP/IP suite uses 224.0.0.0 to 239.255.255.255. The packet is then sent to the network, which makes copies of the packet and sends a copy to each segment with a node that is part of the multicast address. Figure 2-2 is an example of a multicast network.

Figure 2-2 *Multicast Network*



Broadcast

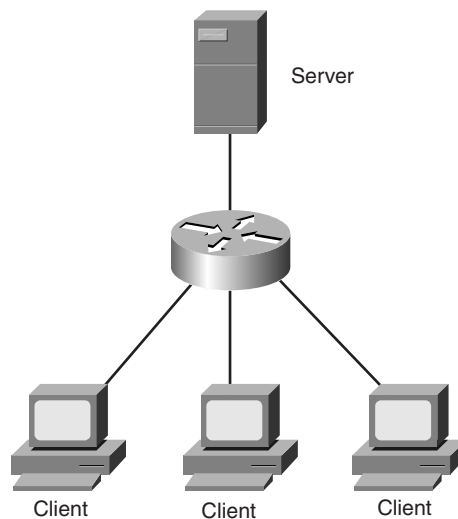
Broadcasts are found in LAN environments. Broadcasts do not traverse a WAN unless the Layer 3 edge-routing device is configured with a helper address (or the like) to direct these broadcasts to a specified network address. This Layer 3 routing device acts as an interface between the local-area network (LAN) and the wide-area network (WAN).

NOTE Broadcasts will traverse a WAN if the WAN is bridged.

NOTE

Ethernet is a broadcast environment in which one device transmits and all other devices see the transmission. Ethernet (broadcast) operation should not be confused with other LAN or WAN broadcasts, where the frame addressed to the broadcast address (a broadcast frame) is copied and forwarded across the network. Figure 2-3 is an example of a broadcast network.

Figure 2-3 *Broadcast Network*



Multimedia broadcast traffic is a much more bandwidth-intensive broadcast traffic type. Multimedia broadcasts, unlike data broadcasts, typically are several megabits in size; therefore, they can quickly consume network and bandwidth resources. Broadcast-based protocols are not preferred because every network device on the network must expend CPU cycles to process each data frame and packet to determine if that device is the intended recipient. Data broadcasts are necessary in a LAN environment, but they have minimal impact because the data broadcast frames that are traversing the network are typically small. Broadcast storms can cripple a network in no time because the broadcasting device uses whatever available bandwidth is on the network.

An example of a data broadcast on a LAN could be a host searching for server resources, such as Novell's IPX GNS (Get Nearest Server) or AppleTalk's Chooser application.

Unlike data broadcasts, which are usually made up of small frames, multimedia broadcasts are typically several megabits in size. As a result, multimedia broadcasts can quickly consume all available bandwidth on a network, bringing a network and its attached devices to a crawl, if not render them inoperable.

Table 2-1 demonstrates the amount of bandwidth that multimedia applications can consume on a network.

Table 2-1 *Multimedia Bandwidth Impact on a LAN (1.5 Mbps* Stream)*

Link Type	Full-Screen, Full-Motion Client/Server Connections Supported (1.5 Mbps Stream)
10 Mbps	6 to 7
100 Mbps	50 to 60
1000 Mbps	250 to 300

*Mbps = megabits per second

For video-conferencing applications, 384 kilobits per second (Kbps) is the recommended maximum bandwidth for uncompressed data streams. Any bandwidth in excess of 384 Kbps typically will not be noticed by end users and could be considered a waste of bandwidth—and in some cases, money. Table 2-2 shows multimedia bandwidth impact on a LAN.

Table 2-2 *Multimedia Bandwidth Impact on a LAN (384 Kbps Stream)*

Link Type	Full-Screen, Full-Motion Client/Server Connections Supported (384 Kbps Stream)
10 Mbps	24 to 28
100 Mbps	200 to 240
1000 Mbps	1,000 to 1,200

LAN Addressing

LAN (or any internetwork) addresses identify individual or groups of devices. Addressing schemes vary depending on the protocol family and OSI layer.

MAC Addresses

Media Access Control (MAC) addresses identify network devices in LANs. MAC addresses are unique for each LAN interface on a device. MAC addresses are 48 bits in length and are expressed as 12 hexadecimal digits. The first six hexadecimal digits, which are administered by the IEEE, identify the manufacturer or vendor and comprise the organizational unique identifier (OUI). The last six hexadecimal digits comprise the interface serial number, or another value administered by the specific vendor. MAC addresses are sometimes referred to as burned-in addresses (BIAs) because they are burned into read-only memory (ROM) and are copied into random-access memory (RAM) when the interface card initializes.

MAC addresses are supported at the data link layer of the OSI model. According to the IEEE's specifications, Layer 2 comprises two components: the MAC sublayer and the logical link control (LLC) sublayer. The MAC sublayer interfaces with the physical layer (OSI model Layer 1), and the LLC sublayer interfaces with the network layer (OSI model Layer 3).

Network Layer Addresses

Network layer addresses identify a device at the OSI network layer (Layer 3). Network addresses exist within a hierarchical address space and sometimes are called *virtual* or *logical* addresses.

Network layer addresses have two parts: the *network* of which the device is a part and the device, or *host*, number of that device on that network. Devices on the same logical network must have addresses with the same network part; however, they will have unique device parts, such as network and host addresses in an IP or IPX network.

For example, an IP address is often expressed as a dotted decimal notation, such as *x.x.x.x*. Each *x* in the address indicates either a network or host number, demonstrated as *n.n.h.h*. The subnet mask determines where the network boundary ends and the host boundary begins.

LAN Topologies

Four LAN topologies exist:

- Star (Hub-and-Spoke)
- Ring
- Bus
- Tree

Star (Hub-and-Spoke) Topology

All stations are attached by cable to a central point, usually a wiring hub or other device operating in a similar function.

Several different cable types can be used for this point-to-point link, such as shielded twisted-pair (STP), unshielded twisted-pair (UTP), and fiber-optic cabling. Wireless media can also be used for communications links.

NOTE

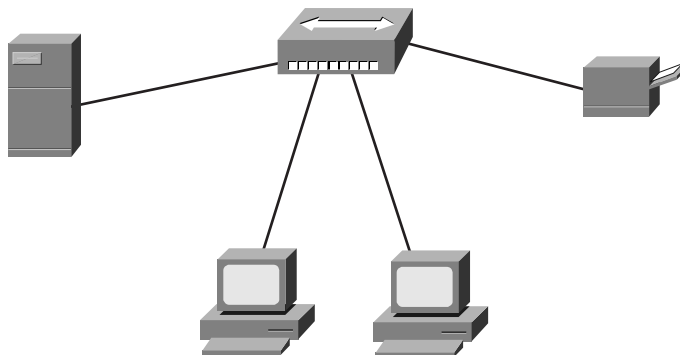
STP is not typically used in a point-to-point configuration. STP is used primarily in the Token Ring environment, where the hubs are called MAUs or MSAUs and the connections from the NIC to the MAU are not really point-to-point. This is because there is a transmit and a receive side, and the transmission is one way. In fact, this is sometimes called a “star-ring.”

The advantage of the star topology is that no cable segment is a single point of failure impacting the entire network. This allows for better management of the LAN. If one of the cables develops a problem, only that LAN-attached station is affected; all other stations remain operational.

The disadvantage of a star (hub-and-spoke) topology is the central hub device. This central hub is a single point-of-failure in that if it fails, every attached station is out of service.

These central hubs, or concentrators, have changed over the years. Today, it is common to deploy hubs with built-in redundancy. Such redundancy is designed to isolate a faulty or failed component, such as the backplane or power supply. Figure 2-4 is an example of a star (hub-and-spoke) topology.

Figure 2-4 *Star (Hub-and-Spoke) Topology*



This example demonstrates a star topology with a file server, printer, and two workstations. If a cable to one of the workstations fails, the rest of the devices are unaffected unless they need to access resources from the “disconnected” device.

Ring Topology

All stations in a ring topology are considered repeaters and are enclosed in a loop. Unlike the star (hub-and-spoke) topology, a ring topology has no end points. The repeater in this case is a function of the LAN-attached station's network interface card (NIC).

Because each NIC in a LAN-attached station is a repeater, each LAN station will repeat any signal that is on the network, regardless of whether it is destined for that particular station. If a LAN-attached station's NIC fails to perform this repeater function, the entire network could come down. The NIC controller is capable of recognizing and handling the defective repeater and can pull itself off the ring, allowing the ring to stabilize and continue operating.

Token Ring (IEEE 802.5) best represents a ring topology. Although the physical cabling is considered to be a star topology, Token Ring is a ring in logical topology, as demonstrated by the following figures. Although physical topology is a physical layer attribute, the media access method used at the data link layer determines the logical topology. Token Ring defines a logical ring and contention, as Ethernet defines a logical bus. Even when attached to a hub, when one Ethernet device transmits, everyone hears the transmission, just as though on a bus. Figures 2-5 and 2-6 are examples of ring topologies.

Figure 2-5 *Ring Topology (Logical)*

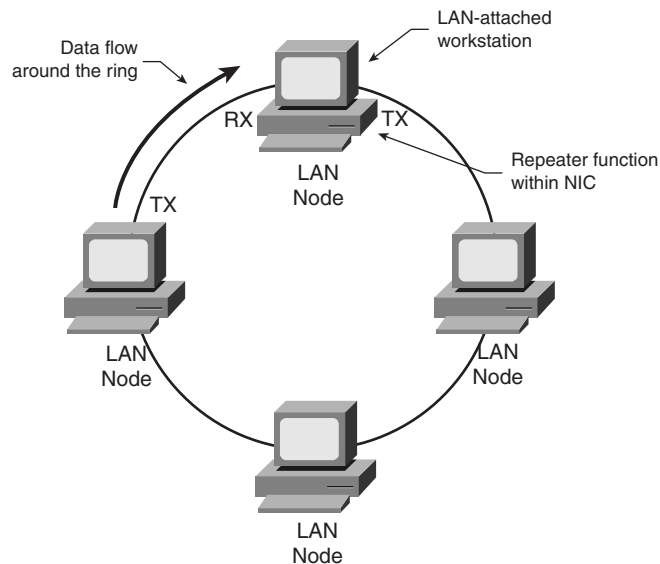
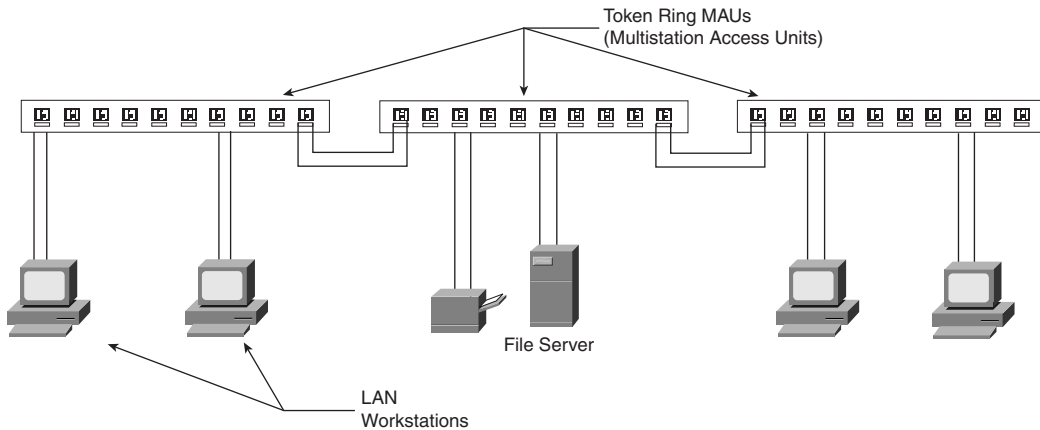


Figure 2-6 Ring Topology



Fiber Data Distributed Interface (FDDI) is another example of a ring topology implementation. Like Token Ring, FDDI rings are physically cabled in a star topology. FDDI stations can be configured either as a single attachment station (SAS) or as a dual attachment station (DAS). SASs are connected to one of the two FDDI rings, whereas DASs are connected to both rings via an A and B port on the FDDI stations and concentrator.

Token Ring and FDDI LANs will be discussed in greater detail in Chapters 6, “Token Ring/IEEE 802.5,” and 7, “FDDI.”

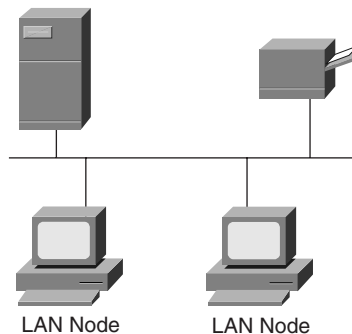
Bus Topology

Sometimes referred to as linear-bus topology, Bus is a simple design that utilizes a single length of cable, also known as the medium, with directly attached LAN stations. All stations share this cable segment. Every station on this segment sees transmissions from every other station on the cable segment; this is known as a *broadcast medium*. The LAN attachment stations are definite endpoints to the cable segment and are known as *bus network termination points*.

This single cable segment lends itself to being a single point of failure. If the cable is broken, no LAN station will have connectivity or the ability to transmit and receive.

Ethernet (IEEE 802.3) best represents this topology. Ethernet has the ability to utilize many different cable schemes. Further discussion of Ethernet and these cable schemes will be found in greater detail in Chapter 3. Figure 2-7 is an example of a bus topology.

Figure 2-7 *Bus Topology*

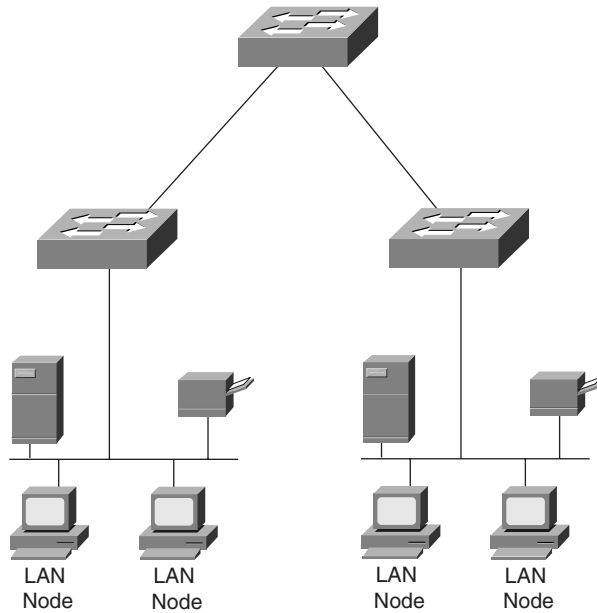


Tree Topology

The tree topology is a logical extension of the bus topology and could be described as multiple interconnected bus networks. The physical (cable) plant is known as a *branching tree* with all stations attached to it. The tree begins at the root, the pinnacle point, and expands to the network endpoints. This topology allows a network to expand dynamically with only one active data path between any two network endpoints.

A tree topology network is one that does not employ loops in its topology. An example of a tree topology network is a bridged or switched network running the spanning tree algorithm, usually found with Ethernet (IEEE 802.3) networks. The spanning tree algorithm disables loops in what would otherwise be a looped topology. Spanning tree expands through the network and ensures that only one active path exists between any two LAN-attached stations. Figure 2-8 is an example of a tree topology.

Figure 2-8 *Tree Topology*



Network Devices

The four primary devices used in LANs are as follows:

- Hubs
- Bridges
- Switches
- Routers

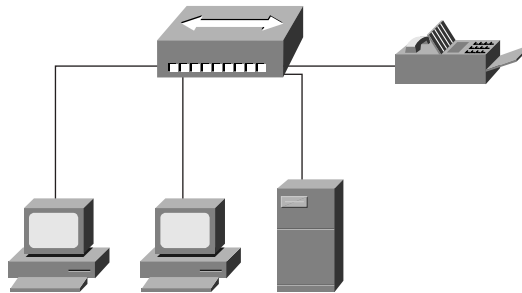
Respective to the OSI model, these devices operate at the following layers:

- OSI Layer 1 (physical)—Hubs, repeaters (hubs are considered to be multiport repeaters)
- OSI Layer 2 (data link)—Bridges, switches
- OSI Layer 3 (network)—Routers

Hubs

Hubs operate at the physical layer (Layer 1) of the OSI model. A hub is used to connect devices so that they are on one shared LAN, as shown in Figure 2-9. Because only two devices can be directly connected with LAN cables, a hub is needed to interconnect two or more devices on a single LAN. The cable termination points are the hub and the LAN device (host).

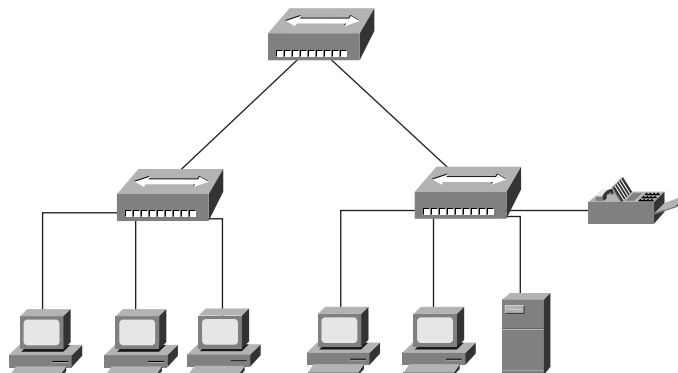
Figure 2-9 *Hub-Based Network*



Ethernet hubs are not “smart” devices; hubs send all the data from a network device on one port to all other hub ports. When network devices are connected via a hub, LAN-attached devices will hear all conversations across the LAN. Each station then examines the message header to determine if it is the intended recipient. If more than one LAN station transmits at the same time, a collision occurs and both stations initiate a backoff algorithm before attempting retransmission. This type of operation is also known as *contention*. All devices attached to the hub are said to be in a *single collision domain*.

Backbone hubs are hubs deployed to connect other hubs to a single termination, or *root*, point. This is known as a *multitiered* design and is illustrated in Figure 2-10.

Figure 2-10 *Backbone, or Multitiered, Hub Network*



Several benefits can be derived from this multitiered design:

- It provides interdepartmental connections between hubs.
- It extends the maximum distance between any pair of nodes on the network.

Intelligent Hubs

Intelligent hubs contain logic circuits that will shut down a port if the traffic originating from that port indicates that bad, or malformed, frames are the rule rather than the exception.

Managed Hubs

Visit www.cisco.com for up-to-date product information and announcements.

Stackable Hubs

Visit www.cisco.com for up-to-date product information and announcements.

Bridges

This section focuses on transparent bridges, which can also be referred to as learning or Ethernet bridges. Bridges have a physical layer (Layer 1), but are said to operate at the data link layer (Layer 2) of the OSI model. Bridges forward data frames based on the destination MAC address.

Bridges also forward frames based on frame header information. Bridges create multiple collision domains and are generally deployed to provide more useable bandwidth. Bridges don't stop broadcast traffic; they forward broadcast traffic out every port of each bridge device. Each port on a bridge has a separate bandwidth (collision) domain, but all ports are on the same broadcast domain.

Bridges were also deployed in complex environments, which is where broadcast storms became such a problem.

Routers were added to the complex bridged environments to control broadcasts. Later, VLANs were devised when switches were deployed in enterprise environments and brought back the old problem of broadcast storms.

NOTE

Bridges, like repeaters, do not modify traffic. Unlike repeaters, bridges can originate traffic in the form of spanning tree bridge protocol data units (BPDUs).

Bridges maintain a MAC address table, sometimes referred to as a *content addressable memory (CAM)* or bridging table, which maintains the following information:

- MAC addresses
- Port assignment

NOTE

Bridges/switches can modify traffic. IP QoS and RIF are examples.

Bridge Operation

A learning bridge examines the source field of every frame it sees on each port and builds up a picture of which addresses are connected to which ports. This means that it will *not* retransmit a frame if it knows that the destination address is connected to the same port on which the bridge saw the frame.

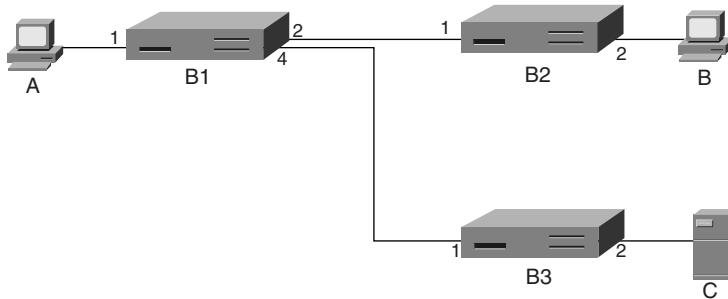
A special problem arises if a bridge sees a frame addressed to a destination that is not in its address table. In this case, the frame is retransmitted on every port except the one on which it was received. This is known as *flooding*.

Bridges also age address table entries. If a given address has not been heard from in a specified period of time, then the address is purged from the address table.

The learning bridge concept works equally well with several interconnected networks, provided that no loops exist in the system. Consider the following simple configuration.

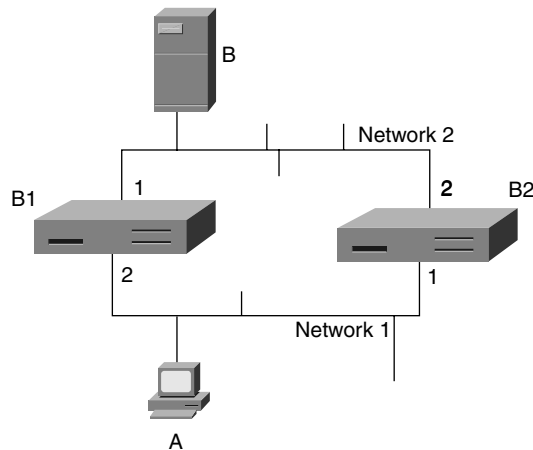
Suppose both stations A and B start up and A attempts to communicate with B (see Figure 2-11). At this point, the following process occurs:

- 1 A frame from A addressed to B reaches port 1 off bridge B1. B1 then learns that station A is connected to port 1, but it knows nothing about station B so it retransmits the frame destined for B on all available ports except port 1.
- 2 Bridge B2 receives the frame. Because Bridge B2 does not know where station B is, it retransmits on all available ports except port 1, causing the frame to reach B and generate a response.
- 3 By examining the incoming frame, bridge B2 knows that A is reachable via its port 1.
- 4 Station B's response reaches B2 on port 2 so that B2 can update its address table with information about the location of B.
- 5 B2 already knows how to get to A, so the response is transmitted on B2's port 1 and reaches B1.
- 6 B1 examines this incoming frame and determines that B is reachable via its port 2.
- 7 Both bridges now know how to send frames to both A and B.

Figure 2-11 *Simple Bridge Network*

The original all-ports broadcast of A's first frame to B ensures that B3 knows how to send to frames to A. An attempt by C to communicate with B results in B3 broadcasting the frame on all ports (except number 2), so the frame reaches B1 on port 4. While B1 forwards this frame to B2, it also learns what to do with frames destined for C.

Unfortunately, this simple and elegant arrangement breaks down disastrously if loops are in the network. Consider the following arrangement. Figure 2-12 is an example of a looped bridge network.

Figure 2-12 *Looped Bridge Network*

Suppose host A has just booted up and wants to communicate with B. A's initial frame will be seen on both Bridge 1's (B1) port 1 and Bridge 2's (B2) port 2, so both bridges know that host A is on network 1. The frame is then transmitted onto network 2 by B1 on port 2 and by B2 on port 1. One of the bridges will transmit it first—suppose it is B1—and then B2 will see a frame from A on its network 2 port 1. It will now update its table as to the location of A and retransmit the frame on network 1. B1 sees this frame and does not know

that it's a duplicate, so it retransmits it on network 2. From there, B2 retransmits it on network 1, and so on indefinitely. Adding a third bridge to this two-network scenario makes things exponentially more complicated.

This is clearly unsatisfactory. Prohibiting loops is an unrealistic target. Practical bridges use a method known as the *spanning tree algorithm* to construct an effective non-looping topology by deciding not to use certain links in the network. It is also possible to reconfigure this network dynamically.

NOTE

Redundant bridges (bridge pairs), added for fault tolerance, cause bridging loops. The ability to reconfigure dynamically helps provide fault tolerance.

Bridges interchange special messages known as *configuration messages*. The spanning tree algorithm (IEEE 802.1) uses BPDUs. The bridge configuration message contains enough information to enable the bridges to do the following:

- Elect a single bridge from among all the connected bridges to be the “root” bridge.
- Calculate the least cost path to the “root” bridge from each bridge.
- For each LAN, identify a “designated bridge” on that LAN that will be used for forwarding frames toward the root.
- Choose a port on each bridge that gives the best path toward the root.
- Select ports to be included in the spanning tree.

The effective topology after construction of the spanning tree is loop free; this is achieved by effectively choosing not to use certain links between bridges. The links are still there and might come into use if the network is reconfigured.

Configuration messages are sent to a special multicast MAC address, meaning all bridges that use the binary SAP value 01000010. Configuration messages are autonomously originated by bridges, but they are *not* forwarded by bridges. A configuration message contains four pieces of information:

- The ID of the bridge assumed to be root.
- The ID of the bridge transmitting the message.
- The cost of the least cost-known path from the transmitting bridge to the assumed root.
- The port number on which the message was transmitted.

A bridge initially assumes itself to be the root, with a path cost of zero. For each bridge port, a bridge will receive incoming configuration messages from other bridges on the LAN connected to that same port. For each port, the bridge will remember the lowest cost

configuration message. The following algorithm describes how a bridge would determine which of C_1 and C_2 is the better configuration message.

```
if( $C_1$ .root_id <  $C_2$ .root_id)  $C_1$ _is_BETTER
else if ( $C_1$ .root_id >  $C_2$ .root_id)  $C_2$ _is_BETTER
else if ( $C_1$ .root_cost <  $C_2$ .root_cost)  $C_1$ _is_BETTER
else if ( $C_1$ .root_cost >  $C_2$ .root_cost)  $C_2$ _is_BETTER
else if ( $C_1$ .tx_id <  $C_2$ .tx_id)  $C_1$ _is_BETTER
else if ( $C_1$ .tx_id >  $C_2$ .tx_id)  $C_2$ _is_BETTER
else if ( $C_1$ .port_id <  $C_2$ .port_id)  $C_1$ _is_BETTER
else if ( $C_1$ .port_id >  $C_2$ .port_id)  $C_2$ _is_BETTER
```

If the bridge configuration message that a bridge receives on any port is better than the bridge configuration message it would transmit, the bridge stops transmitting configuration messages on that port. The bridge uses the new information to recalculate the spanning tree information in the BDPUs configuration messages that it will transmit out ports other than the one from which the new information was learned.

This method details how a network starts up. It is also necessary for networks to be able to reconfigure automatically if a node or link fails or a new node or link comes online. To allow for reconfiguration updates, all stored configuration messages in a bridge are aged. After the age of a configuration message exceeds a certain value, it is discarded and the configuration is recalculated. In the normal course of events, the root bridge periodically transmits configuration messages with an age of zero; receipt of these by bridges causes the bridges to transmit their own configuration messages, also with an age of zero. The time between such messages is called the *Hello Time*.

After the network has stabilized, bridges will issue configuration messages only if they receive such messages or if the age of their internal messages has exceeded the maximum. Configuration messages with age zero can only be transmitted if a configuration message with age zero has been received.

Bridges might not attempt to forward data traffic while the spanning tree is being calculated. In fact, they should not even attempt the “learning” phase until the tree has been defined. This is called the *forward delay*. A special “topology change” flag in a configuration message forces a bridge into the spanning tree calculation mode.

Figure 2-13 illustrates the format of a bridge configuration message, and the following lists gives an explanation of each field.

Figure 2-13 Configuration Message BPDU Frame Format

Protocol Identifier (2 Bytes)
Version (0) (1 Byte)
Message Type (0) (1 Byte)
Flags - Topology Change Advice (1 Byte)
Root ID (8 Bytes)
Cost of Path to Root (4 Bytes)
Bridge ID (8 Bytes)
Port ID (2 Bytes)
Message Age (2 Bytes)
Maximum Age (2 Bytes)
Hello Time (2 Bytes)
Forward Delay (2 Bytes)

- Protocol identifier—Contains the value zero.
- Version—Contains the value zero.
- Message type—Contains the value zero.
- Flags—Signifies one of two events: topology changes or acknowledgements to topology changes.
- Root ID—Defines the bridge that is at the top of the spanning tree.
- Cost of path to root—Defines the accumulated cost from the advertising bridge to the root bridge in the network.
- Bridge ID—Identifies the bridge that generated the BPDU and is used by the algorithm to build a spanning tree.

- Port ID—Defines from which port this BPDU message left the bridge. Other bridges use this to detect and remove loops in a network.
- Message age—Defines the last time the root bridge advertised a BPDU message on which the current network configuration is based.
- Maximum age—Defines the age at which the protocol will remove the information from its database and initiate a topology change by rerunning the spanning tree algorithm. This parameter allows all bridges to age uniformly and to rerun the spanning tree algorithm in parallel.
- Hello time—Serves as the interval in which a bridge advertises BPDUs.
- Forward delay—Defines the length of time that a port will remain in a port state. (Forward delay is discussed later in this chapter.)

A topology change can be advised using a type 128 message with only the first four fields present. Such messages are called topology change notification BPDUs and are defined by IEEE 802.1. They are encapsulated in normal LAN data link layer frames using SAP 01000010 (binary) and are sent to a special multicast MAC address that means “all bridges.”

Latency

Bridges introduce latency, or delay, when forwarding traffic because of the overhead involved. This latency is measured from the moment that the first bit of a frame enters the input port on the device until the time that the first bit of the same frame is forwarded out of the exit port.

NOTE Bridges introduce about 20 to 30 percent loss of throughput for some applications.

Latency has a severe negative impact with some time-dependent technologies, such as voice, video, or mainframe applications.

High levels of latency can result in loss of connections and noticeable video and voice degradation. Routers (OSI Layer 3 network devices) were introduced to overcome the inherent problems of bridging over multiple segments.

NOTE Routers introduce more latency than a bridge does. Routers were introduced to contain broadcasts, one of the other problems associated with flat, bridged networks.

Types of Bridging

Following is a discussion of the four types of bridging:

- Transparent bridging
- Source-route bridging
- Source-route translational, or mixed-media, bridging
- Source-route transparent bridging

Transparent Bridging

Transparent bridging is so named because its operation is transparent to the network hosts. When a host on a remote LAN sends data to a specific destination, it does not look to see where on the bridging LAN the data is. The transparent bridge will read the source frames and forward the data, as discussed earlier.

The major difference between the tables built by bridges and the tables built by routers (Layer 3 devices) is that bridge tables are based on the MAC addresses, whereas routers build their tables based on the network addresses.

Transparent bridges build their tables independently of each other, rather than exchange information like routers. Each bridge learns different MAC addresses by associating the source addresses of transmitted frames with the port on which the frame arrived into the bridge. Each entry in this bridge table has a maximum age associated with it. If this maximum age timer is exceeded—meaning that no traffic has originated from that port within the defined timeframe—the entry is flushed out of the table.

A discussion of bridge states can be found later in this chapter, in the section “Spanning Tree Topology.”

Source-Route Bridging

NOTE

IBM developed source-route bridging. It was later adopted into the IEEE 802.5 (Token Ring) standard.

In a source-route bridged (SRB) network, frames are sent with the complete source-to-destination path included. Source-route bridges check frames for destination information and store and forward as appropriate. The source will make the forwarding choice based on configurable source-route bridging metrics.

In an SRB network, end systems, or hosts, send an explorer frame to the network to find a path from source to destination prior to sending data. The source-route bridges are responsible for adding the path information to these explorer frames and making sure they are passed to and from the appropriate end systems. In addition to passing these explorer frames, source-route bridges also store this routing information in what is called a *RIF cache*. Source-route bridges look into a Token-Ring frame and determine whether routing information exists by checking the routing information indicator (RII) bit. The bridges then add the RII bit and bridge information to the routing descriptor (RD) field, also called the *routing information field (RIF) or RI field*.

Unlike transparent bridges, source-route bridges do not build and maintain tables of MAC addresses and associated ports. Instead, source-route bridges examine the contents of each Token-Ring frame as follows:

- 1 Source-route bridges start by examining the first bit of a Token-Ring frame's source address to see if the value is a zero or a one. This first bit is the RII. The source host of the frame sets the value of the RII.
- 2 If the RII is set to zero, no source-route information exists in the Token-Ring frame.
- 3 If the RII is set to one, source-route information exists within the Token-Ring frame and resides in the RIF.

Three types of explorer frames are found in a source-route bridged network: local explorers, all-routes explorers, and spanning tree explorers.

- Local explorers are used with local source-route bridged networks. Local source-route bridging directly connects two or more Token-Ring networks. Bridged traffic does not cross non-Token media.
- All-paths explorers, as the name implies, take all possible paths on their way to the destination. The amount of traffic generated by all-paths explorers could be considerable in a complex network, which is not good.
- Spanning tree explorers solve the problem of the all-paths explorer by sending packets only to branches in the spanning tree. The network administrator can statically assign which interfaces will forward spanning tree explorer frames and which interfaces will block them. The network administrator can also use the spanning tree algorithm to automatically set a single route explorer. (The spanning tree algorithm and the Spanning Tree Protocol will be discussed later in this chapter.)

Source-Route Translational, or Mixed-Media, Bridging

Source-route translational bridging (SR/TLB) is used when connecting two networks that are running different types of bridging technologies; the most common are Ethernet and Token-Ring. SR/TLB is implemented to perform several functions:

- Overcome MTU and frame format differences between Ethernet and Token Ring.

- Reconcile differences between Token-Ring frames (which contain RIFs) and Ethernet frames (which never contain RIFs) by using Source-route translational bridging.
- Resolve the formatting differences between Ethernet and Token Ring. Token Ring addresses are in non-canonical format; Ethernet addresses are in canonical format.

NOTE

The MTU for Ethernet is 1,500 bytes. The MTU for 4 Mbps Token Ring is 4,550 bytes. 16 M and 100 M Token-Ring is 18.2 KB, or 18,200 bytes.

Source-route translational bridging assures that all these differences are resolved when forwarding frames from Token Ring to Ethernet and Ethernet to Token Ring.

NOTE

The translational, transparent bridge translates only the frame format between many of the IEEE protocols and Fiber Distributed Data Interface (FDDI). It is transparent bridging on all ports.

Source-Route Transparent Bridging

Source-route transparent (SRT) bridging is a bridge that will either source-route bridge or transparently bridge a Token-Ring frame. The RII value makes this determination to either source-route or transparently bridge the frame.

- If the RII value is zero, the frame will be transparently bridged.
- If the RII value is one, the frame will be source-route bridged.

Cisco-Specific Bridging Solutions

Cisco has developed the following five alternative solutions to the previously discussed bridging options:

- Concurrent routing and bridging (CRB)—When CRB is implemented on a Cisco router, specific protocols can be bridged and routed to specific interfaces.
- Integrated routing and bridging (IRB)—Although CRB allows the concurrent routing and bridging of the same protocol on the same routing device, the two never mix. IRB allows bridged and routed traffic of the same protocol to be interchanged. By creating a logical interface, called the Bridge Virtual Identifier (BVI), bridged traffic of a given network layer protocol can be forwarded to a routed interface of the same protocol, and vice versa.

- Virtual rings for multiport source route bridges—Whereas standard Token-Ring bridges have only two ports, Cisco routers can be configured as a multiport source-route bridge by creating a virtual ring within the router. On a multiport source-route bridge, frames from physical interfaces are first forwarded to the virtual ring, and then to another physical interface.
- Remote source-route bridging (RSRB)—RSRB takes the concept of a virtual ring a step further. Instead of forwarding Token-Ring frames from one physical interface to another through a virtual ring, RSRB forwards Token-Ring frames from physical Token-Ring interfaces to interfaces connected to an IP cloud through a virtual ring. The benefit is that this provides a method for performing source-route bridging over a WAN, such as Frame Relay or ATM.
- Data-link switching plus (DLSw+)—DLSw+ is backward compatible with RSRB. DLSw+ performs the same functional tasks that RSRB does, with additional options supported. DLSw+ also supports interconnection of transparent bridging (TB), SRT bridging, SR/TLB, and SDLC-to-LAN conversion (SDLLC) over an IP backbone.

Switches

LAN switches are used to connect a common broadcast domain (a hub). They are also used to provide frame-level filtering as well as dedicated port speed to specific end users. Some switches have limited routing capabilities and can provide Layer 3 routing functions at the most basic level. Some of the major benefits of using switches in a network are higher bandwidth to the desktop and ease of configuration. Switches are being deployed more often to replace hubs and bridges as more bandwidth-intensive applications are being implemented at all levels of an organization.

Switch Operations

The following discussion focuses on Ethernet switches. The switches transfer data on a network by receiving data frames from a source port and forwarding them out to the destination through a different port on the switch based on the frame information. Like transparent bridges, Layer 2 Ethernet switching works by looking at the MAC addressing information in the data frame's header and forwarding the data according to the switch, or Content Addressable Memory (CAM), table information. If the switch looks at the MAC addressing information and still doesn't know from which port to send out the frames, it will broadcast the frames out all of the switch ports. This is known as *flooding*, and it is used to determine the destination. After the destination address is found, the information is added to the switching table.

Switches work by providing dedicated bandwidth per port to an end user or application. Switches allow fewer users in each network segment, and they provide dedicated bandwidth, which is increasingly important with graphics and multimedia applications.

Deploying LAN switches in an existing network environment requires minimal configuration and little or no changes to existing wiring closets, hubs, LAN cabling, or NICs.

Switches allow network users the ability to transfer data traffic in a network environment free of collisions and bandwidth contention. Several types of switching technologies enable quick and scalable network transmission.

Switching Modes

The switch can be configured in a variety of ways to allow certain network services and features to be available within the network.

The modes of LAN switching found today are store-and-forward, cut-through, runt-free, and adaptive cut-through. Some LAN switches can also support the router modes of fast switching and Layer-3 switching.

Store and Forward Switching

This is one of the two common modes of LAN switching. A store and forward switch works by reading and copying the entire data frame into its buffers. Error checking is performed, and the destination address is looked up in the MAC address table. After the switch has determined to which interface the frame should switch, the frame is forwarded to the appropriate destination.

Cut-Through Switching

The other common LAN switching mode is cut-through switching, which allows faster processing than store-and-forward switching. A switch using cut-through switching will copy the destination address and a small portion of the frame to its buffers before checking for the destination address interface in its MAC address table. As soon as the destination is found, the frame is sent out the appropriate port on the switch. Increased switching speed is realized because the cut-through switch does not copy the entire frame to the switch buffers. Cut-through switches enable faster processing by reducing the latency introduced by the switch to a small and constant value—the time it takes to read 6 bytes.

NOTE

Cut-through switching has an inherent danger to it in that the propagation of bad packets, such as runts or frames with an invalid CRC, can occur. To prevent this forwarding of “bad” frames runt-free (read 64 bytes before forwarding) or adaptive cut-through (fallback to store-and-forward if too many errors), modes might need to be implemented.

Fast Switching

Fast switching is the process of copying data frame headers into a memory buffer. You determine the path to the destination host by looking up the destination in the fast-switching cache, building a new frame header/trailer, and forwarding the frame out the appropriate interface. After you determine the destination host path, future data frames will use that switching path, reducing path determination time because the path and outgoing interface have already been established.

NOTE

This is more of a router function than a LAN switch function. Higher-end Cisco switches can perform this function with an installed router blade.

Layer 3 Switching

Layer 3 switching differs from traditional Layer 2 switching by enabling data frames to be switched based on network addressing information. Traditional Layer 2 switching will look at the frames for the MAC address information for the intended destination.

Layer 3 switching can use some routing functions, such as addressing and path determination. Switches can be configured like routers into an addressing mechanism, but they are still bound by a flat-network addressing scheme. Switches that operate at Layer 3 do not support features such as path optimization and load balancing because these features are based on routing processes.

Spanning Tree Algorithm

The spanning tree algorithm is based on the IEEE 802.1 standard, which specifies standards for network management at the hardware level. The spanning tree algorithm is used to ensure that only a single path is selected when using bridges or routers to pass messages—usually in the form of BPDUs between networks—and to find a replacement path if the selected path fails.

The Spanning Tree Protocol (STP) is based on this algorithm and is defined by the IEEE 802.1d standard.

Spanning Tree Protocol (IEEE 802.1d)

As stated earlier, when multiple bridges or switches are interconnected with multiple paths, a looped topology may be formed. A looped topology is often desirable to provide redundancy, but looped traffic is undesirable. Bridged traffic is especially vulnerable to broadcast loops. The Spanning Tree Protocol, IEEE 802.1d, was designed to prevent such loops from being formed. The Spanning Tree Protocol was originally developed for

bridges. Today, it is also applied to LAN switch topologies. By applying the Spanning Tree Protocol to a looped bridged or LAN switch topology, all bridged segments will be reachable. However, any points where loops can occur will be blocked.

Spanning Tree Operation

The Spanning Tree Protocol has four phases of operation:

- 1 Electing a root bridge among a bridge/LAN group
- 2 Calculating the least-cost path to the root bridge/LAN switch by all non-root switches
- 3 Blocking higher cost paths to the root bridge/LAN switch by all non-root switches
- 4 Maintaining and recalculating the spanning tree with BPDUs

After a spanning tree is formed, all bridges and LAN switches know who the root bridge/LAN switch is, what direction the root bridge/LAN switch is in, and what the lowest path cost to the root of the spanning tree is.

When a bridge port or LAN switch port is first activated, it broadcasts BPDUs, with you as the root of the spanning tree. When a bridge or LAN switch receives BPDUs from other bridges or LAN switches, it conducts a spanning tree election to determine which bridge is the root of the spanning tree. Parameters used to determine this spanning tree root bridge include a spanning tree bridge/LAN switch priority number and a MAC address identifying the bridge or LAN switch. Only one root bridge exists in a single spanning tree at any given time.

Spanning Tree Topology

A spanning tree topology consists of the following basic components:

- Bridges/LAN switches
- Bridge/LAN switch segments

A spanning tree topology also consists of the following types of bridges:

- Root bridge/root LAN switches
- Designated bridges/LAN switches
- Non-root bridges/LAN switches
- Non-designated bridges/LAN switches

As previously stated, at any given time only one root bridge/LAN switch exists for the entire spanning tree. A spanning tree election process selects a root bridge/LAN switch.

A designated bridge/LAN switch is the device closest to the root bridge/LAN switch on a given segment. At any given time, only one designated bridge/LAN switch exists for each

segment. A spanning tree election process selects a designated bridge/LAN switch. Each bridge segment must have a designated bridge.

NOTE The non-designated bridge is called the *backup bridge*.

The root bridge/LAN switch is the designated bridge for all segments to which it is attached.

A spanning tree topology consists of the following types of ports:

- One designated port residing on each designated bridge for each bridge/LAN switch segment.
- One root port on every non-root bridge/LAN switch. The root port is the port that provides the most optimal path to the root bridge on a given bridge or LAN switch.

Port States

The IEEE 802.1d specification defines five port states for spanning tree:

- 1 Disabled—This is a unique state for a port. A port that is in a disabled state has either been disabled by the switch because of physical problems or security, or it has been manually disabled by the network administrator.
- 2 Blocking—When a port is in the blocking state, it only listens for BPDUs from other bridges. It does not listen to or save addresses or forward data frames.

NOTE In this state, the bridge assumes that it is the root until it exchanges BPDUs with other bridges.

- 3 Listening—Passing from a blocking state, a port will then enter into a listening state. In this state, a port will listen for frames to detect available paths to the root bridge but will not take source MAC addresses of end stations and place them into the bridge's address table. Also in this state, the bridge will not forward user frames.
- 4 Learning—Upon completion of the listening state, a port will move into a learning state. In the learning state, a port will examine data frames for source MAC addresses and place these in the bridge's address table. Like the listening state, no user data frames are forwarded while the port is in this state.

- 5 Forwarding—After completing the learning state, a port will then be placed into a forwarding state where the bridge will perform its normal functioning. It will learn source MAC addresses and update the bridge's CAM table as well as forward frames through the bridge.

NOTE

Root bridge/LAN switch ports and designated ports are never in a blocking state.

When a bridge or LAN switch port is activated, it normally goes through three spanning tree states: listening, learning, and forwarding. If the port is the highest cost path to the root bridge in a looped topology, it enters the blocking state. By default, all bridge ports go through the first two states: learning and listening. Based on the information they obtain during these states, the interface attains a forwarding or blocking state.

Typically, the spanning tree algorithm takes 50 seconds to calculate a new topology. The transition time for each state is as follows:

- From blocking to listening—20 seconds
- From listening to learning—15 seconds
- From learning to forwarding—15 seconds

Latency, in addition to normal operation, is incurred when the ports go through the different states due to a network change, such as a failed path, addition of a new bridge or switch, or enabling a bridge or switch port. Cisco uses a default value of 15 seconds for the forward delay time, used to measure the time a port stays in a specific state.

CAUTION

In any bridged network, it is important to keep track of the number of times the Spanning Tree Protocol is run. As previously discussed, when each bridge runs the spanning tree algorithm, no user traffic is moved around the network, causing a disruption of service to the end users and their resources. Adding bridges to a network can lengthen the time it takes for the spanning tree algorithm to run its course.

Routers

Routers are not usually active in simple LAN environments because routers are WAN devices. Routers are typically found at the edge of a LAN, interfacing with a WAN. Routers operate at the network layer (Layer 3) of the OSI model. Broadcast containment and security are needed in more complex environments.

Whereas bridges and switches will use the spanning tree algorithm to determine the optimal path to a destination, routers use an algorithm based on the routing protocol that is implemented.

Summary

The three categories of LAN transmission are as follows:

- Unicast—One-to-one transmission
- Multicast—One-to-many transmission
- Broadcast—One-to-all transmission

LAN addressing uses the Layer 2 Media Access Control (MAC) burned-in address (BIA) on the network interface hardware. This address is 48 bits (12 hexadecimal) in length. The first 24 bits, or 6 hexadecimal digits, signify the organizational (vendor manufacturer) identifier as determined by the IEEE. The second 24 bits, or 6 hexadecimal digits, signify a value administered by the specific vendor.

The four primary LAN topologies are as follows:

- Star (hub-and-spoke)—All stations are attached by cable to a central point.
- Ring—All stations are considered repeaters and are enclosed in a loop. Logical configuration is a ring; physical configuration might be a ring or a star.
- Bus—All stations are directly attached to a shared cable segment.
- Tree—All stations are interconnected via several bus networks in a logical extension to the bus topology.

The four primary devices used in LANs include the following:

- Hubs—Hubs operate at the physical layer (Layer 1) of the OSI model and are essentially multiport repeaters, repeating signals out all hub ports.
- Bridges—Bridges create multiple collision domains. Bridges work at the physical layer (Layer 1) of the OSI model and operate at the data link layer (Layer 2). Bridges forward data frames based on the destination MAC address. Bridges utilize the spanning tree algorithm for path determination.
- Switches—LAN switches are essentially multiport bridges. LAN switches are used to connect common broadcast domains (hubs) and to provide frame-level filtering as well as dedicated port speed to end users. LAN switches are also used to create virtual LANs (VLANs). Like bridges, switches use the spanning tree algorithm for path determination.
- Routers—Routers are typically found at the edge of a LAN, interfacing with a WAN, or in more complex LAN environments. Routers operate at the network layer (Layer 3) of the OSI model.

The four types of bridges are as follows:

- Transparent bridges—These create two or more LAN segments (collision domains). They are transparent to end devices.
- Source-route bridging—Frames are sent from the source end device with the source-to-destination route, or path, included.
- Source-route translational, or mixed-media, bridging—These are used when connecting networks of two different bridging types (transparent and source-route) or media types, such as Ethernet and Token Ring.
- Source-route transparent bridging—This bridge will either source-route or transparently bridge a frame depending on the routing information indicator (RII) field.