

Chapter 1 : Receiving Data Switching Modes

The first step in LAN switching is receiving the frame or packet, depending on the capabilities of the switch, from the transmitting device or host. Switches making forwarding decisions only at Layer 2 of the OSI model refer to data as frames, while switches making forwarding decisions at Layer 3 and above refer to data as packets. This chapter's examination of switching begins from a Layer 2 point of view. Depending on the model, varying amounts of each frame are stored and examined before being switched.

Three types of switching modes have been supported on Catalyst switches:

- Cut through
- Fragment free
- Store and forward

These three switching modes differ in how much of the frame is received and examined by the switch before a forwarding decision is made. The next sections describe each mode in detail.

Cut-Through Mode

Switches operating in cut-through mode receive and examine only the first 6 bytes of a frame. These first 6 bytes represent the destination MAC address of the frame, which is sufficient information to make a forwarding decision. Although cut-through switching offers the least latency when transmitting frames, it is susceptible to transmitting fragments created via Ethernet collisions, runts (frames less than 64 bytes), or damaged frames.

Fragment-Free Mode

Switches operating in fragment-free mode receive and examine the first 64 bytes of frame. Fragment free is referred to as "fast forward" mode in some Cisco Catalyst documentation. Why examine 64 bytes? In a properly designed Ethernet network, collision fragments must be detected in the first 64 bytes.

Store-and-Forward Mode

Switches operating in store-and-forward mode receive and examine the entire frame, resulting in the most error-free type of switching.

As switches utilizing faster processor and application-specific integrated circuits (ASICs) were introduced, the need to support cut-through and fragment-free switching was no longer necessary. As a result, all new Cisco Catalyst switches utilize store-and-forward switching.

Chapter 2 : Switching Data

Regardless of how many bytes of each frame are examined by the switch, the frame must eventually be switched from the input or ingress port to one or more output or egress ports. A *switch fabric* is a general term for the communication channels used by the switch to transport frames, carry forwarding decision information, and relay management information throughout the switch. A comparison could be made between the switching fabric in a Catalyst switch and a transmission on an automobile. In an automobile, the transmission is responsible for relaying power from the engine to the wheels of the car. In a Catalyst switch, the switch fabric is responsible for relaying frames from an input or ingress port to one or more output or egress ports. Regardless of model, whenever a new switching platform is introduced, the documentation will generally refer to the "transmission" as the switching fabric.

Although a variety of techniques have been used to implement switching fabrics on Cisco Catalyst platforms, two major architectures of switch fabrics are common:

- Shared bus
- Crossbar

Shared Bus Switching

In a shared bus architecture, all line modules in the switch share one data path. A central arbiter determines how and when to grant requests for access to the bus from each line card. Various methods of achieving fairness can be used by the arbiter depending on the configuration of the switch. A shared bus architecture is much like multiple lines at an airport ticket counter, with only one ticketing agent processing customers at any given time.

Next Figure illustrates a round-robin servicing of frames as they enter a switch. Round-robin is the simplest method of servicing frames in the order in which they are received. Current Catalyst switching platforms such as the Catalyst 6500 support a variety of quality of service (QoS) features to provide priority service to specified traffic flows.

1. Frame received from Host1? The ingress port on the switch receives the entire frame from Host1 and stores it in a receive buffer. The port checks the frame's Frame Check Sequence (FCS)

for errors. If the frame is defective (runt, fragment, invalid CRC, or Giant), the port discards the frame and increments the appropriate counter.

2. Requesting access to the data bus? A header containing information necessary to make a forwarding decision is added to the frame. The line card then requests access or permission to transmit the frame onto the data bus.
3. Frame transmitted onto the data bus? After the central arbiter grants access, the frame is transmitted onto the data bus.
4. Frame is received by all ports? In a shared bus architecture, every frame transmitted is received by all ports simultaneously. In addition, the frame is received by the hardware necessary to make a forwarding decision.
5. Switch determines which port(s) should transmit the frame? The information added to the frame in step 2 is used to determine which ports should transmit the frame. In some cases, frames with either an unknown destination MAC address or a broadcast frame, the switch will transmit the frame out all ports except the one on which the frame was received.
6. Port(s) instructed to transmit, remaining ports discard the frame? Based on the decision in step 5, a certain port or ports is told to transmit the frame while the rest are told to discard or flush the frame.
7. Egress port transmits the frame to Host2? In this example, it is assumed that the location of Host2 is known to the switch and only the port connecting to Host2 transmits the frame.

One advantage of a shared bus architecture is every port except the ingress port receives a copy of the frame automatically, easily enabling multicast and broadcast traffic without the need to replicate the frames for each port.

Crossbar Switching

In the shared bus architecture example, the speed of the shared data bus determines much of the overall traffic handling capacity of the switch. Because the bus is shared, line cards must wait their turns to communicate, and this limits overall bandwidth.

A solution to the limitations imposed by the shared bus architecture is the implementation of a crossbar switch fabric. The term *crossbar* means different things on different switch platforms, but essentially indicates multiple data channels or paths between line cards that can be used simultaneously.

In the case of the Cisco Catalyst 5500 series, one of the first crossbar architectures advertised by Cisco, three individual 1.2-Gbps data buses are implemented. Newer Catalyst 5500 series line cards have the necessary connector pins to connect to all three buses simultaneously, taking advantage of 3.6 Gbps of aggregate bandwidth. Legacy line cards from the Catalyst 5000 are still compatible with the Catalyst 5500 series by connecting to only one of the three data buses. Access to all three buses is required by Gigabit Ethernet cards on the Catalyst 5500 platform.

A crossbar fabric on the Catalyst 6500 series is enabled with the Switch Fabric Module (SFM) and Switch Fabric Module 2 (SFM2). The SFM provides 128 Gbps of bandwidth (256 Gbps full duplex) to line cards via 16 individual 8-Gbps connections to the crossbar switch fabric. The SFM2 was introduced to support the Catalyst 6513 13-slot chassis and includes architecture optimizations over the SFM.

Chapter 3 : Buffering Data

Frames must wait their turn for the central arbiter before being transmitted in shared bus architectures. Frames can also potentially be delayed when congestion occurs in a crossbar switch fabric. As a result, frames must be buffered until transmitted. Without an effective buffering scheme, frames are more likely to be dropped anytime traffic oversubscription or congestion occurs.

Buffers get used when more traffic is forwarded to a port than it can transmit. Reasons for this include the following:

- Speed mismatch between ingress and egress ports
- Multiple input ports feeding a single output port
- Half-duplex collisions on an output port
- A combination of all the above

To prevent frames from being dropped, two common types of memory management are used with Catalyst switches:

- Port buffered memory
- Shared memory

Port Buffered Memory

Switches utilizing port buffered memory, such as the Catalyst 5000, provide each Ethernet port with a certain amount of high-speed memory to buffer frames until transmitted. A disadvantage of port buffered memory is the dropping of frames when a port runs out of buffers. One method of maximizing the benefits of buffers is the use of flexible buffer sizes. Catalyst 5000 Ethernet line card port buffer memory is flexible and can create frame buffers for any frame size, making the most of the available buffer memory. Catalyst 5000 Ethernet cards that use the SAINT ASIC contain 192 KB of buffer memory per port, 24 kbps for receive or input buffers, and 168 KB for transmit or output buffers.

Using the 168 KB of transmit buffers, each port can create as many as 2500 64-byte buffers. With most of the buffers in use as an output queue, the Catalyst 5000 family has eliminated head-of-line blocking issues. (You learn more about head-of-line blocking later in this chapter in the section "Congestion and Head-of-Line Blocking.") In normal operations, the input queue is never used for more than one frame, because the switching bus runs at a high speed.

Shared Memory

Some of the earliest Cisco switches use a shared memory design for port buffering. Switches using a shared memory architecture provide all ports access to that memory at the same time in the form of shared frame or packet buffers. All ingress frames are stored in a shared memory "pool" until the egress ports are ready to transmit. Switches dynamically allocate the shared memory in the form of buffers, accommodating ports with high amounts of ingress traffic, without allocating unnecessary buffers for idle ports.

The Catalyst 1200 series switch is an early example of a shared memory switch. The Catalyst 1200 supports both Ethernet and FDDI and has 4 MB of shared packet dynamic random-access memory (DRAM). Packets are handled first in, first out (FIFO).

More recent examples of switches using shared memory architectures are the Catalyst 4000 and 4500 series switches. The Catalyst 4000 with a Supervisor I utilizes 8 MB of Static RAM (SRAM) as dynamic frame buffers. All frames are switched using a central processor or ASIC and are stored in packet buffers until switched. The Catalyst 4000 Supervisor I can create approximately 4000 shared packet buffers. The Catalyst 4500 Supervisor IV, for example, utilizes 16 MB of SRAM for packet buffers. Shared memory buffer sizes may vary depending on the platform, but are most often allocated in increments ranging from 64 to 256 bytes.

Chapter 4 : Oversubscribing The Switch Fabric

Switch manufacturers use the term *non-blocking* to indicate that some or all the switched ports have connections to the switch fabric equal to their line speed. For example, an 8-port Gigabit Ethernet module would require 8 Gb of bandwidth into the switch fabric for the ports to be considered non-blocking. All but the highest end switching platforms and configurations have the potential of oversubscribing access to the switching fabric. Depending on the application, oversubscribing ports may or may not be an issue. For example, a 10/100/1000 48-port Gigabit Ethernet module with all ports running at 1 Gbps would require 48 Gbps of bandwidth into the switch fabric. If many or all ports were connected to high-speed file servers capable of generating consistent streams of traffic, this one-line module could outstrip the bandwidth of the entire switching fabric. If the module is connected entirely to end-user workstations with lower bandwidth requirements, a card that oversubscribes the switch fabric may not significantly impact performance. Cisco offers both non-blocking and blocking configurations on various platforms, depending on bandwidth requirements. Check the specifications of each platform and the available line cards to determine the aggregate bandwidth of the connection into the switch fabric.

Chapter 5 : Congestion and Head-of-Line Blocking

Head-of-line blocking occurs whenever traffic waiting to be transmitted prevents or blocks traffic destined elsewhere from being transmitted. Head-of-line blocking occurs most often when multiple high-speed data sources are sending to the same destination. In the earlier shared bus example, the central arbiter used the round-robin service approach to moving traffic from one line card to another. Ports on each line card request access to transmit via a local arbiter. In turn, each line card's local arbiter waits its turn for the central arbiter to grant access to the switching bus. Once access is granted to the transmitting line card, the central arbiter has to wait for the receiving line card to fully receive the frames before servicing the next request in line. The situation is not much different than needing to make a simple deposit at a bank having one teller and many lines, while the person being helped is conducting a complex transaction. A congestion scenario is created using a traffic generator. Port 1 on the traffic generator is connected to Port 1 on the switch, generating traffic at a 50 percent rate, destined for both Ports 3 and 4. Port 2 on the traffic generator is connected to Port 2 on the switch, generating traffic at a 100 percent rate, destined for only Port 4. This situation creates congestion for traffic destined to be forwarded by Port 4 on the switch because traffic equal to 150 percent of the forwarding capabilities of that port is being sent. Without proper buffering and forwarding algorithms, traffic destined to be transmitted by Port 3 on the switch may have to wait until the congestion on Port 4 clears.

Head-of-line blocking can also be experienced with crossbar switch fabrics because many, if not all, line cards have high-speed connections into the switch fabric. Multiple line cards may attempt to create a connection to a line card that is already busy and must wait for the receiving line card to become free before transmitting. In this case, data destined for a different line card that is not busy is blocked by the frames at the head of the line. Catalyst switches use a number of techniques to prevent head-of-line blocking; one important example is the use of per port buffering. Each port maintains a small ingress buffer and a larger egress buffer. Larger output buffers (64 Kb to 512 k shared) allow frames to be queued for transmit during periods of congestion. During normal operations, only a small input queue is necessary because the switching bus is servicing frames at a very high speed. In addition to queuing during congestion, many models of Catalyst switches are capable of separating frames into different input and output queues, providing preferential treatment or priority queuing for sensitive traffic such as voice.

Chapter 6 : Forwarding Data

Regardless of the type of switch fabric, a decision on which ports should forward a frame and which should flush or discard the frame must occur. This decision can be made using only the information found at Layer 2 (source/destination MAC address), or on other factors such as Layer 3 (IP) and Layer 4 (Port). Each switching platform supports various types of ASICs responsible for making the intelligent switching decisions. Each Catalyst switch creates a header or label for each packet, and forwarding decisions are based on this header or label.