Cut-Through Ethernet Switching
A Versatile Resource for Low Latency and High Performance

Answering the Challenges of Latency

Today's networks are called on to carry ever-rising volumes of triple-play services and multiprotocol traffic. With more points to monitor and soaring line speeds, the need for business-critical security, compliance, availability, and performance is putting pressure on switching speed. Latency is unacceptable, and businesses cannot afford for any links not to operate at peak performance levels.

Changes to the traditional network design architecture, High Performance Computing (HPC) and increased virtualization in the data center require a faster network. As Layer 2 packet-handling holds the key to lowering latency, discussions always narrow on the two Ethernet switching options that perform packet forwarding: cut-through and store-and-forward. When a Layer 2 Ethernet switch begins a forwarding decision, the series of steps that switch takes to determine whether to forward or drop a packet differentiates cut-through methodology from store-and-forward.

While a store-and-forward switch makes a forwarding decision on a data packet only after it has received the whole frame and checked its integrity, a cut-through switch engages in the forwarding process soon after it has examined the Destination MAC (DMAC) address of an incoming frame.

Both methodologies have their uses, depending upon the application and business considerations. This paper offers some points to consider when deciding on the right approach for a customer data center.

Store-and-Forward—Safer but Slower?

Basically, with store-and-forward switching—a mature technology commonly used in traditional bridges and routers—a switch receives a frame on a port, copies the whole frame into its onboard buffers and stores that frame in its memory buffer until it has been totally received. It then computes a cyclic redundancy check (CRC) and discards the frame if it contains a CRC error or if it is a "runt" (less than 64 bytes including the CRC) or a "giant" (more than 1518 bytes including the CRC). If the frame contains no errors, the LAN switch looks up the destination address in its forwarding, or switching, table and determines the outgoing interface. It then forwards the frame toward its destination.

Less common but a resource to consider is Fragment-Free—or "runtless"—switching. This works similarly to cut-through but stores the first 64 bytes of the packet before sending it on. The logic behind this approach is that most errors—and all collisions—take place during the initial 64 bytes of a packet.

Cut-through architecture results in the very low latency that's vital to High-Performance Computing environments as well as financial trading. Store-and-forward switching requires a larger buffer to accommodate and analyze an entire packet, resulting in higher latency.

Cut-Through Switching for a World In a Hurry

Cut-through technology was developed in the 1990s as a response to issues arising in the 1980s, when networks began to slow due to the rising volume of traffic. Advances in integrated circuit technology streamlined Ethernet switching based on new application-specific integrated circuits (ASICs) and field-programmable gate arrays (FPGAs). These dramatically reduced packet-handling time.

Cut-through switching is not fussy: it forwards a frame almost the (micro)second it has been received, hanging onto it only long enough to read the Destination Media Access Control (DMAC). It performs no error checking because it starts forwarding the frame after the first six bytes.

Fast Forward or Not-So-Fast: Determining Which Is Right for You

The logic is obvious: If an application demands ultra-low-latency (end-to-end latency of about 10 microseconds), then cut-through switching is the right fit. But such factors as port density and cost, as well as the capabilities of the current technology must also be taken into consideration.

In many data centers, latency and jitter (delay variation) are the determining metrics. Even small amounts of latency or jitter can have a severe impact on application performance. In the data center and inside the LAN, where devices are close together, every microsecond counts. And latency is no friend to HPC applications: the more transactions an organization can process in a given time frame, the more revenue it can expect. More time and fewer transactions equals lower revenue.

Holding Back—Also a Wise Strategy?

Ensuring that a packet is free of physical and data-link errors before forwarding may be a smart move under certain circumstances. However, as network throughput increases, the cut-through method is preferred more and more in switch design. Holding packets up while the store-and-forward method performs its checks is often less than ideal for today's impatient world.

However, to be fair, the ingress buffering process of store-and-forward switching provides flexibility to support a diverse mix of Ethernet speeds, starting with 10 Mbps. Store-and-forward simplifies the handling of an incoming frame to a 1 Gbps Ethernet port that must travel to a 10 Gbps interface precisely because the switch architecture stores the entire packet. So it needn't execute additional ASIC or FPGA code to evaluate the packet against an access control list (ACL); the packet is already stored in a buffer for the switch to assess reasons to allow or refuse that frame. Some might argue that store-and-forward reduces error rate, but this may be merely an implementation issue.

The Need for Speed Keeps Growing

Cut-through switching flags but does not get the chance to drop invalid packets. Packets with physical or data-link-layer errors are forwarded to other segments of the network. At the receiving end, a host can then invalidate the Frame Check Sequence (FCS) of the packet and drop the packet.

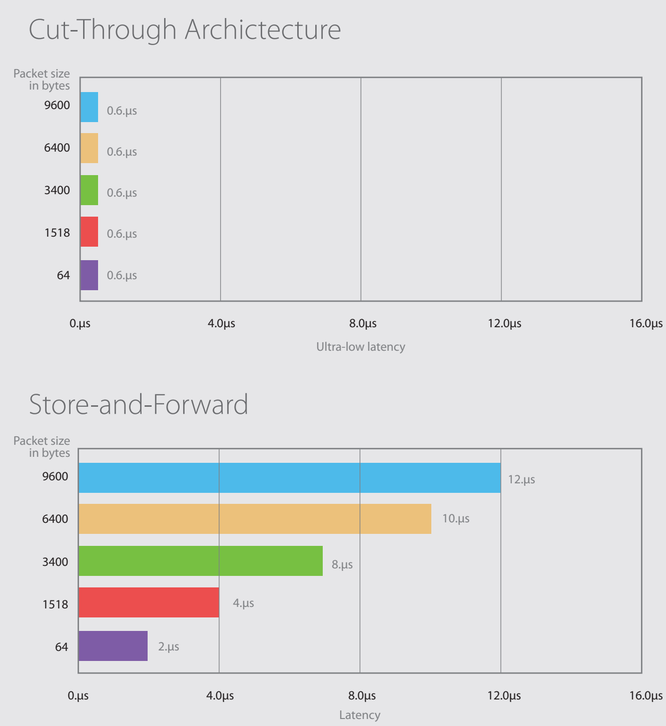
Both store-and-forward and cut-through Layer 2 switches base their forwarding decisions on the DMAC address of data packets. They also learn MAC addresses as they examine the source MAC (SMAC) fields of packets as stations communicate with other nodes on the network.

A cut-through switch theoretically makes its forwarding decision the moment it has looked up the DMAC address of a data packet, without having to await the rest of the packet. However, in reality, it can afford to wait until a few more bytes have arrived. In fact, newer cut-through switches can collect more information about an incoming packet and make a better-informed decision—even approaching the "wisdom" of the store-and-forward methodology. For example, Cisco's advances in ASIC design now enable improved cut-through functions that, along with load balancing and other capabilities, allow low-latency switching while preserving the inspection benefits of store-and forward. .

Director xStream™ and iLinkAgg xStream™: Low Latency Cut-Through Architecture for High-Performance Computing

These two Net Optics solutions are engineered for the ultra-low latency that enables monitoring solutions for today's and tomorrow's needs.

The following diagram illustrates the architecture advantages of cut-through switching



Read about cut-through switching in the Net Optics e-book, *The Top 5 Ways to Enhance Your Cisco Environment.* Download it at www.netoptics.com/elibrary

When Cut-Through Switching Is Not the Ideal Approach

Certainly, store-and-forward switching delays the time it takes for the frame to get from source to destination. That's because it waits to forward a frame until it has received the entire frame and checked it for errors, comparing the last field of the datagram against its own frame-check-sequence (FCS) calculations. So that additional time is spent ensuring that the packet is purged of physical and data-link errors. Invalid packets are dropped while a cut-through device would simply forward them on. Also, a store-and-forward switch can perform ingress buffering for the flexibility to support any mix of Ethernet speeds.

Regarding Director xStream's behavior in cut-through mode: Disabling CRC and jumbo packet forwarding causes changes in behavior. The reason is that packets are already being transmitted from the output port by the time the chip can determine that a CRC error or jumbo packet condition exists. At that point, the chip truncates transmission of the packet. In the case of a packet with a CRC error, most of the packet has already been transmitted. A jumbo packet is truncated after 1524 bytes have been transmitted. It is recommended that a user operate with CRC forwarding and jumbo packet forwarding enabled to prevent confusion.

Each organization must determine the choices and mixes of store-and-forward or cut-through switching that best fit its unique environment. For HPC applications, the decision is fairly easy: reduce latency to the lowest possible levels in order to stay competitive. HPC enterprises include entertainment, oil and gas exploration, aerospace and biosciences, financial data mining, government research and climate/weather simulation.

Find out more about how Net Optics helps you make the right decision about gaining the benefits of cut-through switching in your network. Visit www.netoptics.com or contact us at (408) 737-7777.

Customer First!