Victor Lama
Fabric Specialist G500
28 February 2011

An Introduction to Priority-based Flow Control

Some History for Context

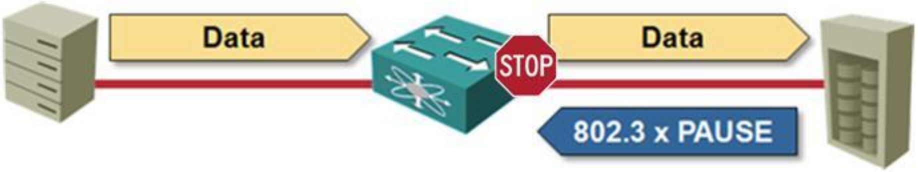
In 1997, the IEEE 802.3 work group ratified an extension known as 802.3X. This extension included an annex (31B) that provided specifications regarding link-level flow control. In short, the standard describes a method by which a receiving station transmits an Ethernet PAUSE MAC control frame to a sender that is causing its input buffers to be overrun. The goal is to suspend the transmission of frames for a period of time so that the receiver is not forced to drop them.

In an overrun condition, a PAUSE frame will be transmitted back to the sender, requesting an immediate cessation of all traffic flow; the key word here being “all.” The duration of the silence is measured in multiples of time, known as quanta, with a single quantum equal to the time it takes to transmit 512 bits. The idea is to make the duration of the PAUSE a function of the speed of the link and not an absolute quantity of time. As a result, the PAUSE time for a 10G link is shorter than for a 1G link. The link in question could be between Ethernet switches or between a NIC and a switch port.

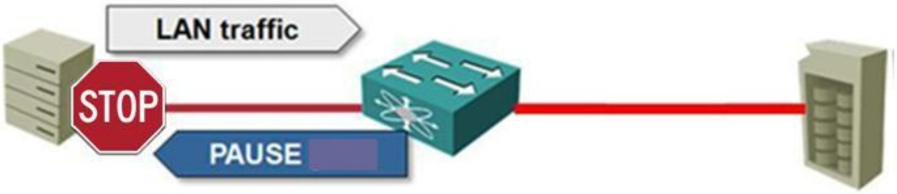
At the time of the 802.3X specification’s release, there were varying views from the industry’s leaders as to the original intent of its creators. Regardless, the major Ethernet appliance vendors agreed that the PAUSE MAC control frame was either inapplicable to their Ethernet switching product line (Cisco and HP) – given the shared memory and non-blocking nature of their switch hardware’s architecture – or was altogether inappropriate for deployment on all but perhaps an edge port that faces a single device.

The figures below give a visual of the 802.3X PAUSE semantic for flow control and its hop-by-hop propagation through a network. Remember, the PAUSE frame is a link-level technique, not an end-to-end flow control solution.

When the server overloads the iSCSI storage array with the data it’s sending, the storage array sends a PAUSE frame back to the switch.



The switch stops sending data to the storage array after receiving the PAUSE frame, causing the data sent by the server to accumulate in the switch's internal buffers until the switch has to tell the server to cease transmitting data by sending its own PAUSE frame.



Deploying the PAUSE flow control method on a link that connected two core devices was considered by all vendors to be a clumsy practice that can lead to unnecessary congestion on a crucial link that would not have otherwise experienced any congestion at all. Moreover, it was clear that the Quality-of-Service/Class-of-Service (QoS/CoS) architecture would be severely disrupted by the PAUSE frame, causing mission critical traffic with a high priority to suffer unnecessary delay and jitter. With QoS, it was a zero-sum-game: either use the PAUSE frame and stop all traffic from flowing or deploy a QoS solution.

As a result, vendors like Cisco disabled link flow control by default and only allowed PAUSE frames to be received, but not sent. According to the specification, an implementation can be in compliance without sending PAUSE frames. There was a consistent view that flow control should be relegated to the transport layer, like TCP, or to the application itself when the transport layer protocol (UDP) did not have any flow control mechanism to offer.

In fact, the loss of some packets in a particular flow is considered necessary to trigger the somewhat crude flow control mechanism in TCP, as well as for the Adaptive Queue Management technique used by Random Early Detection (RED) to signal congestion to TCP. The loss of a single TCP segment results in the reduction in size of the sliding window by 50%, which results in less traffic being sent and lower link utilization. As more frames are dropped, the sliding window will eventually decrease to a size of 1, which is referred to as Silly Window Syndrome.

Priority-based Flow Control – IEEE 802.1Qbb

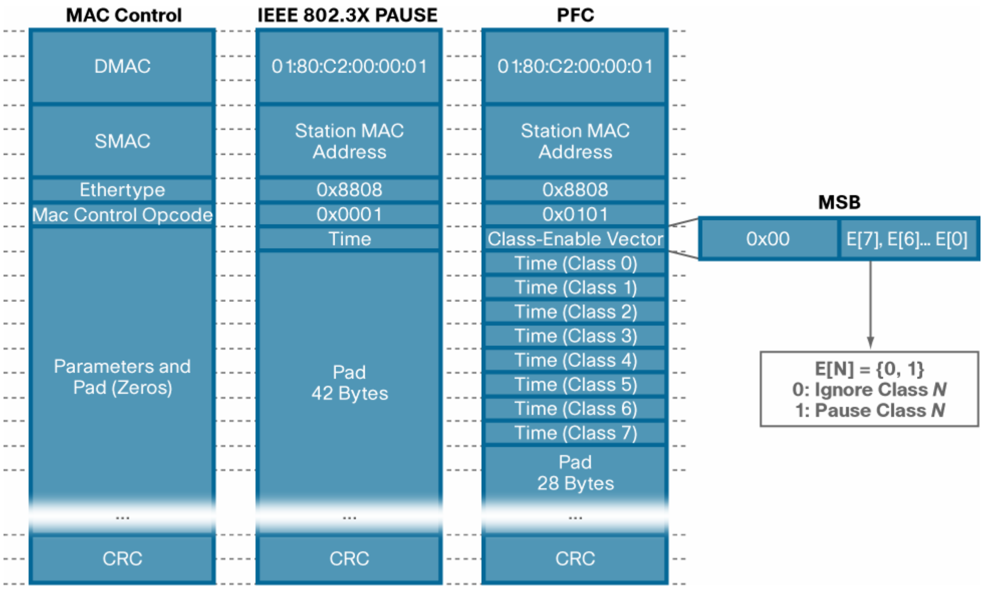
With the advent of FCoE, a requirement emerged to create a transport mechanism for storage traffic that was lossless, just like Fibre Channel (FC). FC is considered reliable because both sides have a pre-shared knowledge of the recipient's buffer capabilities and fluctuating availability. It uses a buffer-to-buffer credit system that ensures that the sender only sends the fixed number of frames that the receiver advertises to the sender upon link initialization – and nothing more. As a buffer becomes available on the receive end of an FC link, an R_RDY frame is sent back to the sender, allowing it to send its next series of frames. This ensures that the receiver's input buffers do not get pummeled with more traffic than the port can handle.

With classic Ethernet, there is no guarantee of successful delivery of a frame injected into the switched domain. Packets *do* get dropped, even when QoS is deployed – albeit with QoS, that situation is more predictable and better managed than it is without it. Nonetheless, this is not enough for FCoE; the transport must be lossless to maintain the integrity of the stored data. Therefore, the INCITS T11 BB5 work group's FCoE specification defines a storage traffic transport model in which the lowest 3 layers of the FC protocol stack (FC0-2) are replaced by a **modified** Ethernet transport that is **lossless**, while preserving the semantics and requirements of FC's Upper Layer Protocols (ULP).

To reiterate, the challenge is to create a network transport that can support various traffic types, each with its own set of transit requirements; for FCoE that requirement is the ability for an FC target or intervening device to receive every frame sent to it without failure. This led to the rebirth of the all-but-abandoned PAUSE control frame method of flow control. **But this time the mechanism has some built-in granularity that allows for flow control to be applied to only certain classes of traffic, as defined by the 802.1p CoS primitives.**

There is a common misconception that the PAUSE frame is meant to stop IP traffic and allow the FCoE traffic to be sent, but in fact it is the other way around. Applying a PAUSE frame to temporarily stop the transmission of FCoE traffic ensures that FCoE frames are not dropped – delayed a bit, perhaps, but not dropped, thereby creating a lossless transport path for that class of traffic. In some vendor implementations, that class-of-service level defaults to CoS 3. The other classes of traffic that do not fall under the lossless rubric will continue leveraging their QoS/CoS semantics to ensure reliability.

The figure below displays the differences in the format of the legacy PAUSE frame with that defined in IEEE 802.1Qbb. Note how the PFC frame now has fields targeting different traffic classes.



A PAUSE in Time Saves…Data!

Note, there's no value in sending a PAUSE frame after all the interface buffers are already occupied. Therefore, the flow control mechanism has to predict when those buffers will be full and transmit the PAUSE frame ahead of time. To ensure timely transmission, certain considerations have to be made.

For the following exercise, picture two devices, S and R. S is the data sender and R is the data receiver. R sends a PAUSE control frame back to sender S when its buffer space reaches a critical threshold. That threshold is defined by taking the following things into consideration:

- The MTU of the interface on R that is sending the PAUSE frame to S. Imagine that a PAUSE frame is ready for transmission at device R (data receiver) at the very moment that the first bit of a frame carrying the full MTU size begins engaging the transceiver logic. The PAUSE frame will have to wait for that packet to serialize before it can be sent. Meanwhile, S continues sending traffic and occupying more of R's buffer space.
- The time it takes for the PAUSE frame to transit the link. This is known as propagation delay. Until it gets to the other end and is acknowledged, S will once again continue transmitting data.
- The response time of S. This is defined as the time it takes for S's internal logic to process the PAUSE frame. The PFC definition caps this as 60 quanta, or 3,840 bytes on a 10G link.
- Transceiver latency on both S and R, which is negligible for SFP+ deployments, but pretty significant when using 10GBase-T, which can account for up to 12,800 bytes of delay.
- The MTU of the interface on S that is receiving the PAUSE frame. Once S acknowledges receipt of the PAUSE frame and is finally ready to comply, it can only do so at packet boundaries. So, once again, imagine that the PAUSE frame is processed by S at the very time that the first bit of a packet carrying a payload with the interfaces MTU has begun engaging the sender's logic. It will have to wait for that packet to be sent before S stops transmitting.

All these timeframes can be calculated given the speed of the link, the MTU sizes, the distance between ends, and the speed at which data travels on a "wire," either copper or fiber. The sum of these timeframes equates to the total amount of buffer space that must be available on R when it sends its PAUSE frame. This is referred to as R's receive threshold.