Technical Brief

Fabric Operating System 9.0
Fabric Notifications

Abstract

Fabric Notifications provides a way to engage end devices (hosts and storage) in automatically remediating congestion issues and response time diagnosis. The goal of Fabric Notifications is to provide end devices with "additional data" that cannot be extracted nor inferred from regular I/O activity. Congestion is an issue, but persistent intermittent errors are equally impactful and potentially more challenging to diagnose. End devices do not know how their behavior is impacting other devices on the SAN. Essentially, each device functions as if it "owns" the network. An end device that requests too much data results in fabric flooding, the return of buffer credits stopping for some reason, or a pathway becoming marginal; in these frequently encountered cases, sufficient resolution by end devices is possible if they are notified. These impairments typically cause fabric congestion, which can apply backpressure on a large number of flows. Storage traffic throughput is optimized by preventing congestion. The avoidance of application performance degradation is paramount and ultimately the goal of Fabric Notifications.

Fibre Channel (FC) is a long-standing proven network technology that implements flow control mechanisms to gain a virtually lossless high-performance storage network. Flow control can potentially pose congestion challenges due to traffic characteristics and misbehaving devices if not addressed. Brocade has addressed and continues advancing its technology to solve these challenges. This technical brief discusses some of these situations, particularly MPIO marginal paths, link integrity issues, oversubscription, and credit stall. Broadcom storage networks address congestion via Fabric Notifications by notifying pertinent attached devices, which can then take evasive action.

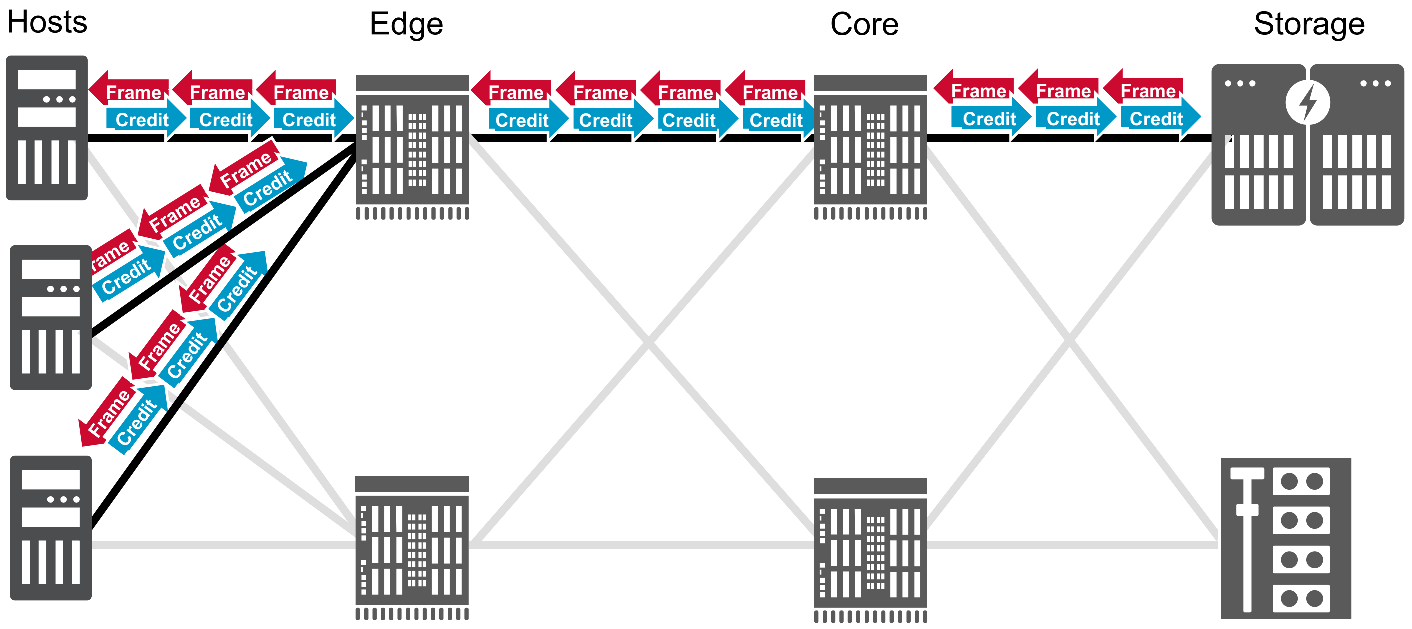
The Goal

In any network, lossless, unencumbered, freely moving frames are perfection. Fibre Channel network technology approaches this perfection and continues to be the superior storage networking technology after many proven years of implementation. Fibre Channel networks are ultrahigh bandwidth, ultralow latency, and lossless—all characteristics critically essential for high-performance host-to-storage communications. Fibre Channel is a ubiquitous technology that is considerably simpler to deploy, easier to manage, and more reliable than other network types. Fibre Channel is lossless; if it were not, it would result in either data loss or added I/O retransmit time, and neither is acceptable.

Creating a congestion-free and outage-free network is the goal. When congestion occurs, it can be pervasive, sometimes affecting 100s of unrelated flows. Fibre Channel networks can be elusive to troubleshoot, visualization of flows can be difficult, and end devices are unaware of SAN problems. Within the SAN, Brocade has Flow Performance Impact (FPI) to detect non-optimal conditions. Brocade has introduced a hardware, software, and management solution for achieving real-time congestion reduction and elimination that is called Fabric Notifications.

In collaboration with end devices, Fabric Notifications solves fabric performance issues. Fabric Notifications and end devices provide each other insight not ordinarily available. By identifying useful data from various sources, status can be collected, evaluated, and disseminated to interested devices, allowing for faster and sometimes automatic problem resolution. End devices can employ the necessary response and recovery mechanisms. Fabric information is useful for end devices, and end devices have useful information for the fabric and their peer end devices. Fabric Notifications plays a crucial role in collecting and disseminating information among interested and related devices.

Figure 1: Freely Moving, Lossless, Credit-Based, Flow-Control FC Network



Fabric Notifications addresses four issues: link integrity, congestion (oversubscription and credit stall), and SCSI command delivery failure. Each case is described in the sections that follow.

Link Integrity

The impact of questionable components along a SAN path is severe and frequently leads to application degradation, crashes, outages, and lost revenue. The integrity of the link between switch ports and an end device is vital to proper operation. Finding and correcting "dead" components is not overly tricky; on the other hand, finding and curing "sick" components is challenging. Link integrity symptoms frequently manifest fabric-wide rather than at the actual problem location. Moreover, link integrity from a switch port to a device port cannot be detected by the fabric itself; errors are detected by the receiver and must then be reported to the fabric.

Devices, cables, SFPs, HBAs, drivers, or patch panels with erratic, unstable, or marginal behavior are not easy or quick to pinpoint. For example, intermittent connectivity and bit errors caused by damaged or dirty cables or marginal but working SFPs are challenging to identify. Cyclic redundancy check (CRC) and invalid transmission word (ITW) are link-level errors that lead to poor performance and long retry times; furthermore, these errors do not cause nor influence a Multipath IO (MPIO) failover. A fabric switch does not disable a potentially faulty port because there is no awareness of an alternate operational path; A and B fabrics are entirely isolated by design. Ironically, hosts and storage are knowledgeable of alternate routes but not their outgoing link integrity issues.

Congestion

Congestion occurs because the rate of frames entering the fabric exceeds the rate of frames exiting the fabric. When downstream frames do not egress at a rate equal to or faster than the ingress rate, frames eventually fill the buffers. Buffers are a tool used to ensure lossless behavior, but they are not the cause of congestion. Credit-based flow control stops returning credits before buffers overflow, and additional ingress frames are blocked at the transmitter. Oversubscription, credit stall, and lost credits are the most common reasons for congestion. An impaired or credit-stalled device upstream causes ISL flows to slow down or halt. If an ISL accommodates many flows, the backpressure caused by a single slow flow inadvertently affects the other flows. Congestion increases response time and reduces application performance because data is not moving consistently or efficiently.

Oversubscription occurs when a device requests more than it can process and is resolved with Fabric Notifications (congestion notifications). A credit stall occurs when a device stops returning credits and is resolved via Fabric Notifications (congestion signals). Lost credits occur with transmission errors and are recovered using credit recovery. Each impairment has a unique signature and resolution.

The following are causes of congestion:

- Oversubscription, which occurs when more frames are arriving than can be processed
- Credit stall, which occurs when a device stops returning credits, effectively bringing the link to a standstill
- Lost credits, which occurs when physical errors damage frames or the credit response and effectively reduces the capacity of the link

The critical difference between oversubscription and credit stall is that with oversubscription, credits are returned as fast as possible, whereas with credit stall, credits are not returned.

Oversubscription

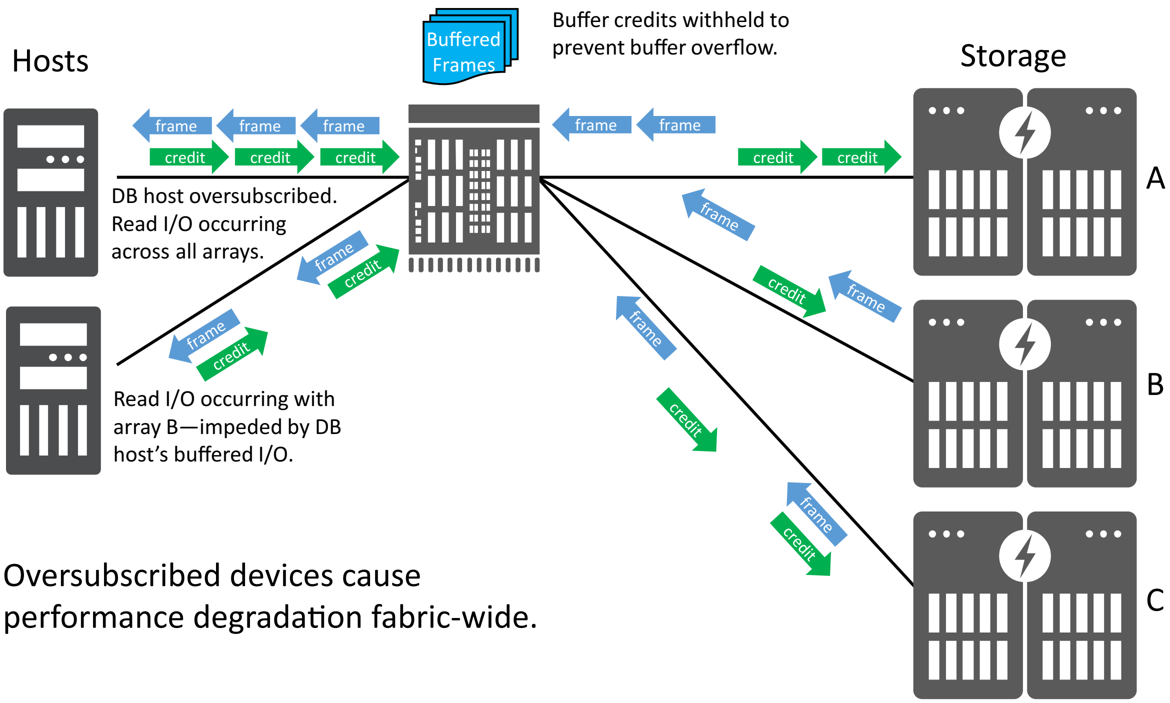
Oversubscription occurs when the incoming data rate is faster than the outgoing data rate, causing a condition in which the receiving device returns credits at a slower pace than necessary to run at full throughput. Oversubscription is not caused by buffer starvation; instead, the egress rate is lower than the aggregate ingress traffic load queued on one or more ports. An insufficient number of buffer credits slows data transmission, causing upstream backpressure. For example, the aggregate data from multiple sending ports exceeds some port's ability along the path to accommodate it. When oversubscription impedes multiple flows, a fabric-wide problem could occur.

F_Ports are fabric ports, which are ports on a director or switch. N_Ports are node ports, which are HBA or storage ports. F_Ports connect to N_Ports. E_Ports are expansion ports, the ports used for the inter-switch link (ISL). E_Ports connect to E_Ports (switch to switch).

Oversubscription can occur under the following circumstances:

- A receiving device port has less bandwidth than the sending device port.
- A receiving device port has less bandwidth than the aggregate of all sending device ports.
- An ISL has less bandwidth than the sending device port.
- An upstream ISL has less bandwidth than a downstream ISL.

Figure 2: Oversubscription

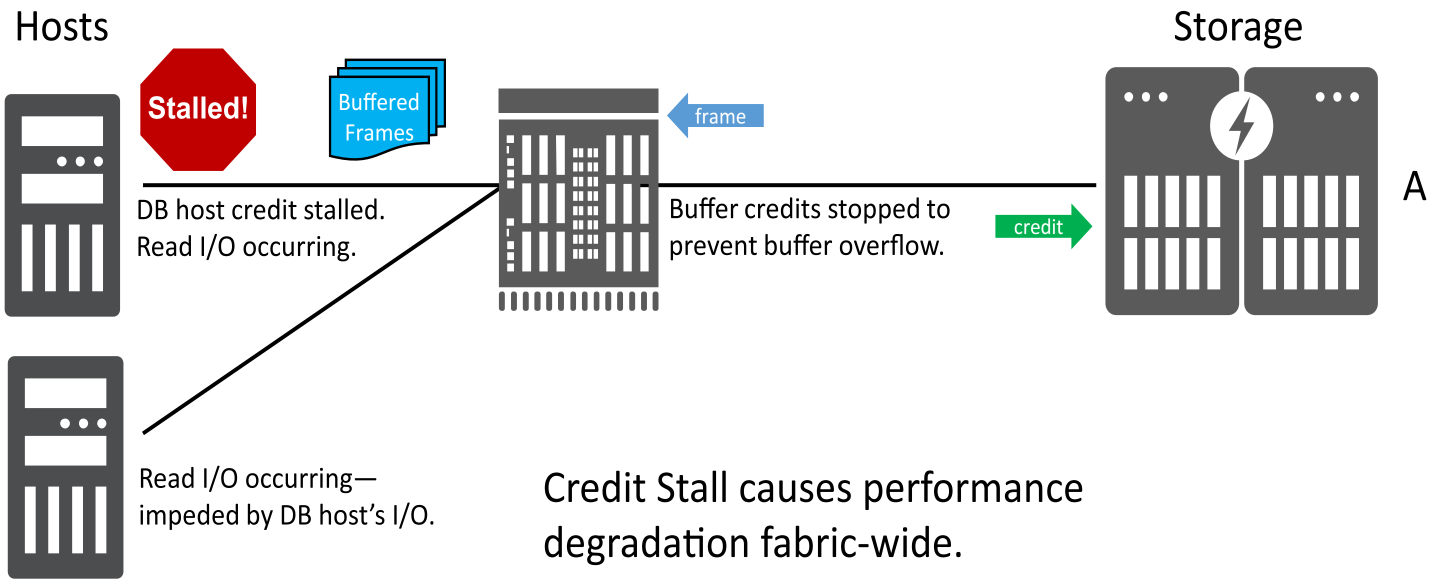

Hosts
Buffered Frames
Buffer credits withheld to prevent buffer overflow.
Storage
frame
credit
DB host oversubscribed. Read I/O occurring across all arrays.
Read I/O occurring with array B—impeded by DB host's buffered I/O.
A
B
C
Oversubscribed devices cause performance degradation fabric-wide.

Credit Stall

Credit stall is an extreme case where the receiving device stops returning credits for extended periods (250 ms or more). When a receiving port stops returning buffer credits to a sending port, data can no longer be sent. There are many reasons why this could occur: driver defects, device malfunctions, loss inflight of all buffer credits, or an algorithm that explicitly stops returning credits for whatever reason (for example, a full queue or resource contention).

A switch port attached to a credit-stalled device impedes all upstream flows, potentially becoming a fabric-wide problem. When traffic queued to leave a switch port stops because the end device has credit stall, the traffic flow coming into the fabric consumes that egress buffer and propagates the stall upstream toward the source. Eventually, the source is signaled to stop sending data into the fabric.

Figure 3: Credit Stall


Hosts
Stalled!
Buffered Frames
frame
Storage
A
DB host credit stalled.
Read I/O occurring.
Buffer credits stopped to prevent buffer overflow.
credit
Read I/O occurring—impeded by DB host's I/O.
Credit Stall causes performance degradation fabric-wide.

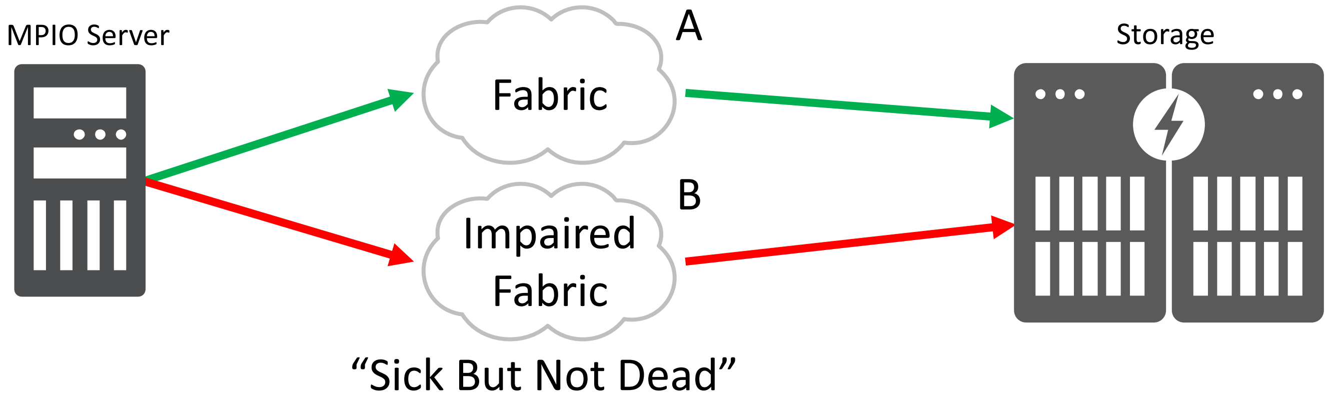
MPIO

Accepted SAN best practice is to construct two parallel autonomous fabrics, traditionally called "A" and "B." Intentionally, no cross-connections are made between the two fabrics, which is referred to as an *air gap*. Fabrics connect to different HBAs on the host and to different controllers on storage. This redundant architecture has a tremendous amount of availability. It is of concern how hosts and storage drive data across these two independent fabrics, especially in light of fabric conditions from the end devices' perspective.

Multipath IO (MPIO) systems address hard errors and effectively fail over to redundant paths. These systems can also effectively deal with congestion using an advanced load-balancing algorithm such as least queue, least blocks, or service time. However, MPIO systems struggle to address intermittent errors, which significantly slow down application I/O due to the sequential nature of most application operations. See Figure 4. It is these persistent, intermittent problems that Fabric Notifications is intended to address. If a pathway is impaired and MPIO continues to send data into the fabric, the result is congestion. Eventually, backpressure is exerted on the sending HBA port. How will MPIO deal with the backpressure? Will it weigh traffic across the pathways (least queue length) or restrict the traffic across both paths to the slowest rate (round-robin)? Furthermore, sending traffic into an impaired fabric is likely to cause blocking of other flows as well.

The path selection function determines the mechanism used by the OS to optimize the selected I/O path. Most commonly, MPIO defaults to round-robin across paths. By definition, round-robin balances traffic equally across each fabric regardless of its ability to deliver data; therefore, this method is not optimal. A debilitated pathway is devastating to performance on the entire SAN. The recommendation is least queue length, which improves both resiliency and performance. The least queue length recommendation specifically addresses congestion introduced by higher latencies on one path compared to the other paths.

Figure 4: MPIO with a Marginal Path



SCSI Command Delivery Failure

SCSI command delivery failure results in a lengthy timeout and application disruption. Dropped SCSI commands can be caused by various fabric conditions: routing changes, congestion, configuration changes, fabric timeouts, and more. The end device does not know expediently if a SCSI command will not be delivered; it must wait until the timeout. On the other hand, in most cases the fabric does know immediately when a SCSI command fails to be delivered. If the end device were to be notified, it could take immediate actions to recover.

Fabric Performance Impact (FPI)

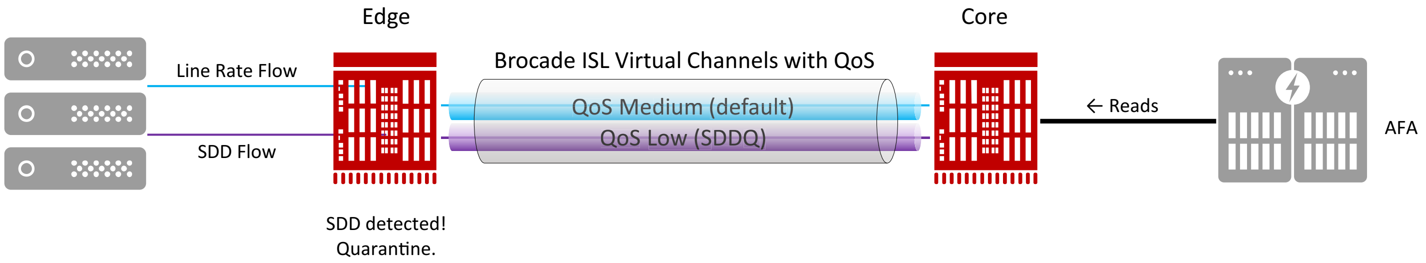
It is essential to understand the basics of Fabric Performance Impact (FPI) before moving onto Fabric Notifications.

Buffer credits are Fibre Channel's flow-control mechanism and are vital to fabric performance. A variety of conditions can negatively affect buffer credits and flows.

Oversubscription is a common characteristic of lossless flow-controlled networks. Other issues that involve buffer credits include driver defects, which can cause a device to exhibit credit-stall behavior, and buffer credits lost in transmission by unstable optics, dirty optics, or damaged cables. All these conditions impact fabric and application performance.

A single misbehaving device can bottleneck portions of a storage network, putting mission-critical applications and service-level agreements (SLAs) at risk. FPI policies specify preset latency severity levels with an intelligent detection algorithm. When a severity level is breached, obstructing flows are quarantined, and SAN administrators are notified via Brocade's Monitoring and Alerting Policy Suite (MAPS). FPI identifies *perpetrator* flows that are hindering the free and efficient transport of other *victim* flows. An identified perpetrator flow is routed to a Brocade QoS low-priority autonomous virtual channel to prevent further impeding of victim flows. Quarantine of perpetrator flows is called Slow Drain Device Quarantine (SDDQ). FPI cannot correct misbehaving devices, but it identifies and limits their impact.

Figure 5: FPI SDDQ



The Brocade® Fabric Notifications Solution

Fabric Notifications is a mechanism that provides end devices with more information about events in the fabric. They include notifications regarding link integrity issues, delivery notification issues, and congestion issues. The Fabric Notifications architecture allows devices to participate at a level that suits their needs and technical evolution. Notifications essentially tell an end device, "You are sending too much into the fabric" or "Beware, there is a problem ahead—slow down or switch paths." The end device is made aware of a problem and can act to initiate remediation.

Fabric Notifications was developed to optimize I/O behavior and avoid impaired paths by notifying devices of current fabric conditions. The Brocade® Fabric Notifications solution combines hardware (Brocade FC switching ASIC), software (Fabric Operating System), and management (MAPS and SANnav™) to form a SAN-wide solution for impairment detection, notification, and remediation. Problems are logged, including operational information concerning affected flows, location, timestamp, and reason for easy debugging and troubleshooting.

Fabric Notifications uses two types of notifications: hardware-based Congestion Signal primitives and software-based Fabric Performance Impact Notifications (FPINs).

Congestion Signal Primitives

Congestion Signal primitives are ASIC-based and are supported only on Brocade Gen 7 platforms. Congestion Signal primitives are optical codes that are sent over the link between directly connected Fibre Channel devices. Congestion Signal primitives are extremely fast and notify peer Gen 7 platforms of degraded performance due to oversubscription or credit-stall conditions. Switching hardware intelligently detects sudden congestion situations and reacts instantly by signaling the attached physically connected port.

Extended Link Service Notifications

Extended Link Service (ELS) is a communication protocol used between Brocade Fabric Operating System (FOS) platforms for exchanging capabilities, service registration, notifications, and other vital control traffic. The Fabric Notifications solution augments hardware Congestion Signal primitives with software FPIN notifications for congestion, peer congestion, link integrity, and SCSI command delivery. Fabric Notifications is based on Fabric OS and is supported on Gen 6 and Gen 7 platforms running FOS 9.0 or later.

Hosts, switches, and storage can generate ELS notifications upon detecting an impacting event. Any fabric member that detects a problem on a switching platform (F_Ports and E_Ports) or an end device (N_Ports) may send a notification. No matter where the ELS notification was generated and which port it arrived on, it is processed the same.

Fabric Notifications and MAPS work in tandem. MAPS rules for congestion, SCSI command discard, and link integrity trigger corresponding FPIN ELS notifications. FPIN notifications are sent to all in-zone end devices that have registered to receive that particular notification type. Notifications are not dependent on zoning type, whether traditional or peer; the operation is the same with no advantage or disadvantage. Additionally, Fabric Notifications sends FPIN notifications to all peer switches, and the peer switches forward notifications to their registered in-zone end devices. FPIN notifications are restricted to devices that:

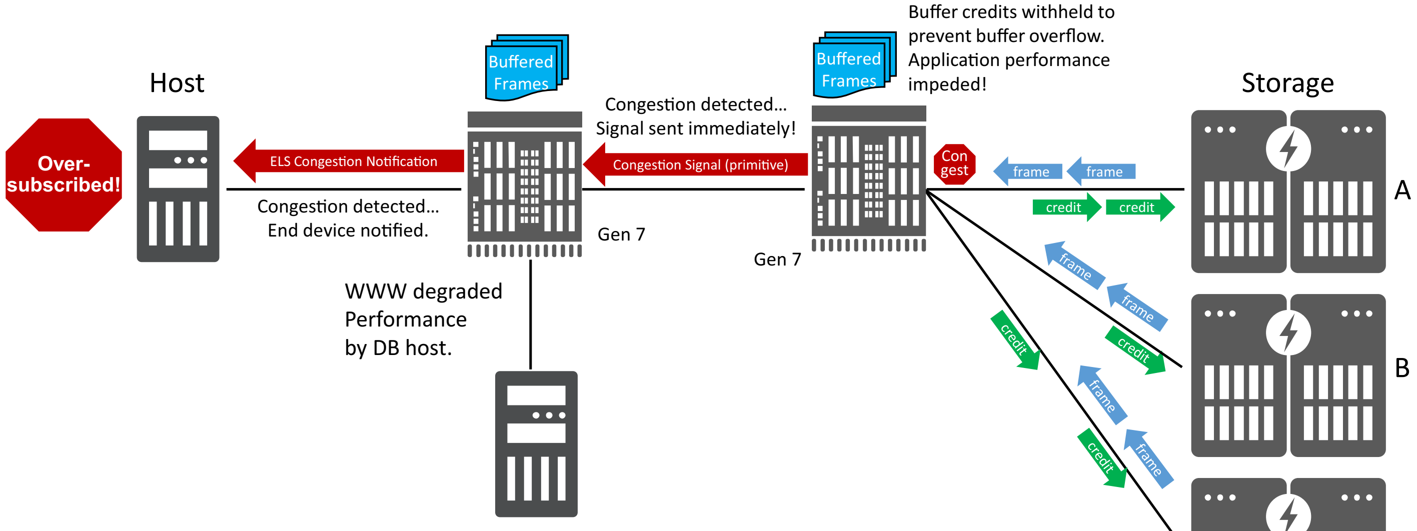
- Support FPIN notifications
- Are registered to receive a particular notification type
- Are experiencing the notification condition
- Are an in-zone peer device

Congestion Signals

Congestion signals are an immediate feedback mechanism indicating that transmission resources are becoming consumed on a link. Congestion Signal primitive signals are immediately sent upon detection, within microseconds (µs). Congestion signals are a link-by-link feature only. Brocade Gen 7 Fibre Channel ASICs support direct congestion signaling between platforms.

In turn, if the condition causes MAPS to trigger, the receiving switch uses FPIN Congestion Peer notifications to other registered in-zone end devices for potential I/O queuing optimization and remediation.

Figure 6: Congestion Signal and Notification for Oversubscription



ELS Congestion notification indicates to the end device that they are impacting the fabric.

Congestion Notifications

FPIN Congestion notifications are valuable information for end devices that can optimize I/O scheduling, for example, slowing transfer rates or issuing serial read I/Os. The ELS Congestion notification is the software equivalent of the Congestion Signal primitive and is sent to end devices that support notifications. An ELS Congestion notification for oversubscription means that too much data was requested, and the fabric could not accommodate the request without congestion. An ELS Congestion notification for credit stall indicates to the end device, "You are stuck, fix the situation." In general, Congestion notifications indicate why long exchange completion times may be occurring.

Congestion Peer Notifications

FPIN Congestion Peer notifications are sent to registered in-zone peers of end devices that are experiencing congestion. There are a variety of remedies that peers can leverage to relieve congestion. The peer's port may have auto-negotiated faster than the destination port; the peer could limit its data rate to match that of the destination. Peers could choose alternate pathways to circumvent a flow that is potentially causing congestion. Paths could be weighted based on Congestion Peer notifications to proportionally deliver traffic over multiple paths in a weighted fashion.

Link Integrity Notifications

MPIO drivers receive Link Integrity notifications and manage path selection. When MPIO is connected to an impaired path, those affected MPIO hosts get notified so they can take action. End devices decide their best course of remediation, such as an immediate failover to an alternate path within the MPIO environment, a change of I/O rate, or a change of queuing algorithm.

The following are two common link integrity problems and their associated source:

- Slow host application performance can be traced to cyclic redundancy check (CRC) errors on a storage port.
- Slow array performance can be traced to host port invalid transmission words (ITWs).

The cabling infrastructure and optics between the end device and the fabric are frequent sources of unreliable communications, but how would the sending device know that the receiver is experiencing a marginal connection? A complete solution requires device ports to notify switch ports of receive-side errors that the fabric would otherwise be unaware of. Links are two-way streets, and FPIN Link Integrity notifications enable identifying degraded paths in each direction.

Notification device to switch (N_Port to F_Port): If an HBA or storage port sees incoming errors, it notifies the fabric that its connection is erratic. The device port sends an FPIN Link Integrity notification to the switch port; additionally, the FPIN is distributed to other in-zone devices. The fabric logs notifications, so the SAN administrator knows that it occurred and when.

Notification switch to device ports (F_Port to N_Port): Switch ports experience link errors from an HBA or storage port when there are faulty electronics, marginal optics, dirty connectors, or damaged cables, causing Link Integrity notifications to be sent to the HBA or storage port to flag a transmission problem. In turn, the HBA or storage port forwards notifications to the MPIO driver, which can distinguish between a device failure and a physical path failure. The end device's administrator uses Link Integrity notifications to troubleshoot, and link integrity is used by MPIO to select the best path.

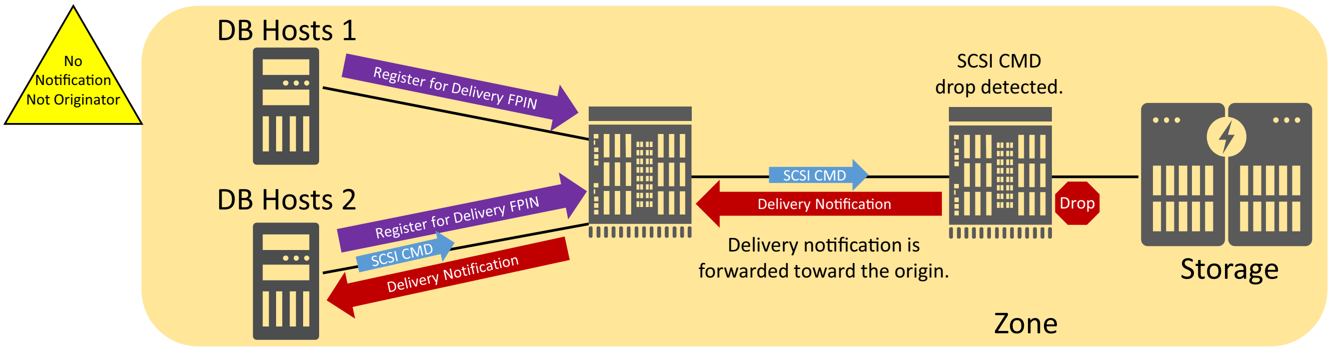
Registered in-zone end devices must be notified of link integrity problems. MAPS F_Port alerts for CRC and ITW errors are translated into Link Integrity FPINs and sent to the end devices. Notifications integrate with the OS device manager, and notifying initiators and targets of link integrity issues along a data path enhances debugging and recovery efforts from multiple perspectives.

SCSI Command Delivery Notifications

When an end-device-originated command drops in transit, the application is likely to time out, and the error recovery process is initiated. Application timeouts vary in length; nonetheless, it can be a significant amount of time, and recovery may not be entirely straightforward depending on the application and I/O in progress. If the end device were to be notified of the SCSI command delivery failure, the application might recover much quicker.

When a fabric has discarded a SCSI command or status frame, Fabric Notifications notifies the initiator of the failure by sending an FPIN Delivery notification. FPIN is used to mitigate application timeouts by immediately telling the end device of the failure. No matter if the command is dropped by an ISL or end device connection, the originator is notified. Now, the end device has the opportunity to retry the command expeditiously. The notification commences an immediate SCSI I/O failure without waiting for a timeout and avoids potential application disruption.

Figure 7: FPIN SCSI CMD Delivery Notification



SCSI CMD Delivery notification indicates to the originating device that there was a delivery failure.

Summary

This technical brief describes how Fabric Notifications provides a way to engage end devices (hosts and storage) in automatically remediating congestion issues and prolonged response times. The goal of Fabric Notifications is to offer end devices "additional data" that cannot be extracted nor inferred from regular I/O activity. Common yet sometimes elusive SAN issues in Fibre Channel flow-controlled networks are examined: MPIO impaired paths, oversubscription, credit stall, link integrity, and SCSI command/status delivery failure. MPIO is unaware of impaired pathways, and the end device does not know that it is flooding the network, making conditions worse. Fabric Notifications makes end devices aware of deleterious fabric conditions so that they can enact some form of intelligent mitigation.

Troubleshooting can be tedious and challenging even though Brocade has all the tools needed to troubleshoot issues. FPIN provides end devices with the capability to adapt to fabric conditions that they otherwise would not be aware of in real time without guesswork on top of performance-based load balancing. Also, FPIN is required because MPIO alone is unaware of fabric conditions. Fabric-wide, oversubscribed and credit-stalled devices can impede flows unassociated with those actually causing the impediment; these are victim flows. Link integrity is another troublesome and common issue because of the sheer number of HBAs, optics, cables, and panels. Brocade fabrics can easily detect and log connectivity issues to and within the fabric. However, once data is transmitted out a fabric port, the end device must report incoming errors to the fabric; otherwise, the fabric remains unaware. The fabric cannot report marginal links and intermittent errors to the SAN administrator if it is not aware of them. Lastly, dropped SCSI command and status frames cause lengthy timeouts and costly application disruptions, which can be mitigated by notifying the end device and preempting a timeout.

The Brocade Fabric Notifications solution addresses each of these issues through standards and a technology ecosystem of innovation. Various notifications with problem-specific descriptors are sent to or received from the attached end devices. In some cases, they are propagated to other directors and switches to notify peer end devices. Fabric Notifications compliments MAPS technology and was introduced by Broadcom in Fabric OS 9.0.

Broadcom reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design. Information furnished by Broadcom is believed to be accurate and reliable. However, Broadcom does not assume any liability arising out of the application or use of this information, nor the application or use of any product or circuit described herein, neither does it convey any license under its patent rights nor the rights of others.

The product described by this document may contain open source software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, to view the licensing terms applicable to the open source software, and to obtain a copy of the programming source code, please download the open source disclosure documents in the Broadcom Customer Support Portal (CSP). If you do not have a CSP account or are unable to log in, please contact your support provider for this information.

BROCADE®
A Broadcom Company