Proactively Improve Performance With Resilient SAN Fabrics

By Steve Guendert, Ph.D.

Modern Fibre Channel (FC) Storage Area Network (SAN) fabrics, both Fibre Channel Protocol (FCP) and Fibre Connection (FICON), have evolved from simple data transport mechanisms to much more complex infrastructure. Fabrics containing multiple hosts across a wide spectrum of operating systems and hundreds of storage ports are common today. I/O levels and data traffic volumes, particularly across the core of a SAN fabric, are much higher.

Fabric usage has also changed significantly. In 2013, there are many more High Availability (HA) requirements and more complex workloads. Hypervisors and virtualized hosts in significant quantity (such as in a Linux on System z implementation) make it more difficult to isolate application problems when application performance becomes an issue. Storage virtualization exacerbates this with its own unique I/O requirements.

All this has a serious impact on storage, and particularly SAN fabric problem determination. There are now more entities to manage, including storage volumes, Logical Unit Numbers (LUNs), hosts, storage arrays, virtual machines, etc., and, therefore, more things that can go wrong. As a result, the operational environment is much more difficult to manage than even a few years ago. Rogue, or poorly behaving devices, have more impact on production environments than previously. All the innovation in workloads and storage infrastructures has generated new behaviors and a corresponding difference in SAN fabric traffic patterns and management demands.

The result of all this change is that the user is likely to see increasing issues with application performance. These issues seem to be associated with storage performance, but can be difficult to pinpoint and correct. Faulty or improperly configured devices, misbehaving hosts and faulty or substandard FC media can significantly impact the performance of FC fabrics and the applications they support. In most cases, these issues can't be corrected or completely mitigated in the fabric itself; the behavior must be addressed directly. However, with the proper knowledge and capabilities, the fabric can often identify and sometimes mitigate or protect against the effects of these misbehaving components to provide better fabric resiliency.

Here we will discuss three aspects of SAN fabric resiliency:

1. HA five 9s architecture and designing for redundancy
2. Detecting abnormal behavior in external components (typically, servers/hosts or storage devices) that can negatively impact the SAN fabric so you can identify and fix the faulty device
3. Mechanisms that protect the SAN fabric from adverse effects caused by a faulty component, including one or more actions you can invoke automatically using a switch when faulty behavior is detected. This can contain and isolate the impact of the misbehaving component in the fabric. This should be considered a temporary measure. Ultimately, the faulty or improperly configured component must be addressed to resolve the problem completely and permanently.

Creating Five 9s Availability for SAN Fabrics

Massive amounts of data are created, transmitted and stored every day. Such data—whether in the form of financial transactions, online purchases, customer demographics, correspondence, spreadsheets or any number of business applications—is the livelihood of businesses across the globe. When it comes to customer transactions, it's imperative that none is lost due to an IT system failure. Users demand near-100 percent system and infrastructure availability. This is no longer a luxury; it's a necessity. Mission-critical applications and operations require truly reliable services and support, especially for their important I/O traffic.

HA is valuable to all businesses, but to some, it's more crucial than others. Deploying HA must be a conscious objective; it requires time, resources and money. HA is used to ensure constant connection of servers to storage networks and storage devices and a reliable data flow; there's also a premium to pay when dealing with the Total Cost of Acquisition (TCA) of HA equipment.

However, the Internet has emphasized that HA equals viability. If companies lack reliable, available HA solutions for the continuing operation of their equipment, they lose money. If a company's server fails, customers are apt to click over to a competitor. If mission-critical computers involved in manufacturing are damaged through machine failure, inventory may come up short and schedules could be missed. If a database application can't reach its data due to I/O fabric failures, seats might not get filled on flights, hotel room reservations might go to a competitor or credit card transactions might be delayed—costing many thousands, sometimes even millions, of dollars in damage to the company's bottom line.

Storage networking is an important part of this infrastructure. Because of their dependency on electronic devices, storage networks can fail. It may be due to software or hardware problems, but failures do occur. That's why, rather than taking big risks, businesses running strategic processes in their computer environments will integrate HA solutions into their operations.

Real-time systems are now a central part of our lives. Reliable functioning of these systems is of paramount concern to the millions of users who depend on these systems daily. Unfortunately, some systems still fall short of user's expectations of reliability. What is really meant by the term five 9s HA in storage networks?

Reliability and Availability

Let's clarify a few terms. "Reliability" and "availability" often are used interchangeably, but they aren't the same. Reliability is about how long something works before it breaks. Availability is about how to keep running even if something does break. Five 9s availability is difficult to achieve. It's not just a matter of building hardware but how that hardware is deployed in the data center that determines if five 9s availability can be achieved.

For example, the industry claims dual director-based fabrics can provide five 9s availability. Maybe, maybe not. Can a user, such as a financial institution, broker or airline

The Internet has emphasized that HA equals viability.

reservation system, survive for long on one-half the bandwidth if one of their two fabrics fails? Usually not. Figure 1 depicts this configuration.

To achieve five 9s availability, a user must architect the hardware, fabrics and bandwidth across the fabrics to achieve this high expectation. The configuration in Figure 1 potentially provides five 9s availability from a redundant fabric perspective. Each of the two deployed fabrics must run at no more than approximately 45 percent busy. That way, if a failure occurs and one fabric must failover to the remaining fabric, the remaining fabric will have the capacity to accept and handle the full workload for both fabrics. Users will often create these fabrics so the data flow across each fabric could be shifted to the other fabric without any issues. However, over time, entropy will occur. Storage devices, Channel Path Identifiers (CHPIDs) and Host Bus Adapter (HBA) ports are added (or removed) in an asymmetrical fashion, leaving more data traffic on one fabric than another. In a similar scenario, application and data growth occur that drive both fabrics to run at above 45 percent utilization. If one fabric had to failover to the other, then the remaining fabric wouldn't have enough excess bandwidth to accept and successfully run the failed fabric's workload.

The question that isn't asked often enough when attempting to deploy HA fabrics is: "Beyond redundant fabrics, how much bandwidth can I lose and continue to meet my Service Level Agreements (SLAs)?"

Worldwide, the most common deployment of mainframe FICON fabrics is shown in Figure 2. Since a Path Group (PG) has eight links, two links (redundancy) are routed to each of four fabrics per PG. Since mainframe channel cards and open system HBAs contain four or fewer ports, it requires at least redundant channel cards or HBAs to provision the eight paths shown in Figure 2. This lets users deploy redundancy for HA fabrics at yet another level of their infrastructure.

Risk of loss of bandwidth is the motivator for deploying fabrics such as those shown in Figure 2. Here, the four fabrics limit bandwidth loss to no more than 25 percent if a fabric were to fail and couldn't be fully recovered by the other fabrics. In this storage networking architecture, each fabric can run at no more than about 85 percent busy, so if a failover occurs, the remaining fabrics can accept and handle the full workload without overutilization. Even with entropy, it would take a tremendous amount of growth to overwhelm the total availability and bandwidth capabilities of this deployment.

By now you understand why so many users in mainframe environments deploy three, four or eight fabrics. In this way, when a single fabric fails, there's sufficient capacity to handle the additional workload from the failed fabric and continue to operate at full capability—a truly five 9s environment. So availability isn't just hardware. It's the customer's ability to get all their work done even if something failed.

Five 9s director-class, FC switching devices have component redundancy built in everywhere.

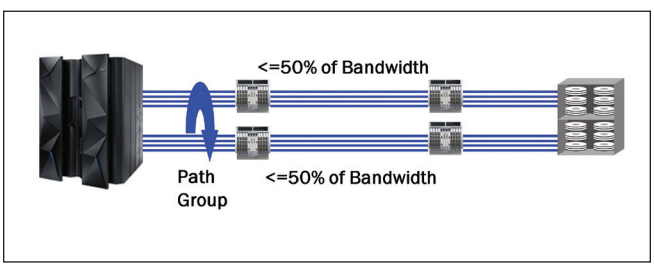

Figure 1: What Happens If You Lose 50 Percent of Your Bandwidth Capacity?

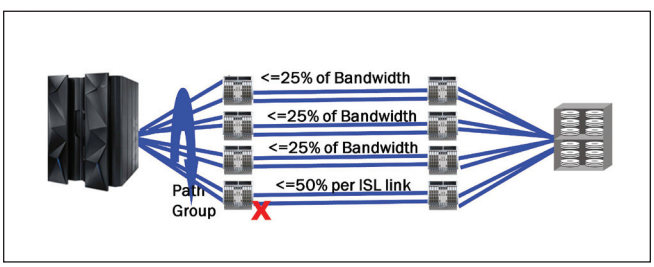

Figure 2: Eight-Way Pathing With Four FICON Fabrics

Each switching vendor and storage vendor selling infrastructure wants to paint the best possible picture of the devices they're selling. Unfortunately, there's no one authority on what devices are two 9s, three 9s, four 9s or five 9s. Vendor marketing departments always emphasize the best possible characteristics of a device, including availability, whether or not there's proof to support the claims. A user must look deeper into the engineering and manufacturing of a device to get a sense of that device's capability to deliver five 9s availability. Warning signs to consider include if a device has:

- A single fan
- A single fan connector (attached to the backplane so if it's damaged, the chassis might need replacement)
- A single I/O timing clock; all I/O must be timed and timers can get out of sync and go bad
- Control and connectivity blades with numerous components; the more components, the potential for failures
- Poor control over I/O frame flow
- An active backplane (hundreds of electronic components attached to their backplanes).

One assumes that switch vendors will do all the correct engineering and have all the proper manufacturing processes so they will create a climate of reliability and availability in their product set. The storage network, as the piece in the middle, is crucial to HA environments. It's important for the user to understand:

- How to design their storage network for five 9s HA, if required
- What to look for in the switching device's engineering and design to ensure HA requirements are being met.

Detecting Abnormal Behavior in a SAN Fabric

There are two common classes of abnormal behavior that originate from fabric components:

1. **Faulty media (fiber optic cables and Small Form-Factor Pluggable (SFPs)/optics):** Faulty media can cause frame loss due to excessive Cyclic Redundancy Check (CRC) errors, invalid transmission words and other conditions. This may result in I/O failure and application performance degradation.
2. **Misbehaving high-latency end devices (hosts or storage):** End devices that don't respond as quickly as expected and cause the fabric to hold frames for excessive periods of time are referred to as slow draining devices. This can result in application performance degradation or, in extreme cases, even I/O failure.

Common examples of moderate device latency include DASD arrays that are overloaded and hosts that can't process data as fast as they request it. Severe latencies are caused by badly misbehaving devices that stop receiving, accepting or acknowledging frames for excessive periods. The bottleneck detection feature discussed in the article, "Troubleshooting Performance Problems With DASD Array Host Adapters," in the June/July 2013 issue of *Enterprise Tech Journal* (available at http://entsys.me/u4ptm) discussed this in detail.

FC switches can't correct bad node behavior or faulty media; they can only attempt to alert and compensate for it. Ultimately, the problems must be addressed in the host or target devices or media where they actually occur.

Faulty Media

In addition to high-latency devices, faulty media often cause fabric problems. The most common component to fail in a SAN is an SFP (optics). Faulty media can also include bad cables, extension equipment, receptacles, patch panels, improper connections and more. Media can fault on any port type (E_Port or F_Port) and fail, often unpredictably and intermittently, making it even harder to diagnose. Faulty media involving F_Ports impact the device attached to the F_Port and devices communicating with this device. Failures on E_Ports can have an even greater impact. Many flows (host/target pairs) can simultaneously traverse a single E_Port. In large fabrics, this can be hundreds or even thousands of flows. In the event of a media failure involving one of these links, it's possible to disrupt some or all the flows using the path.

Severe cases of faulty media, such as a disconnected cable, can result in a complete failure of the media, which effectively brings a port offline. This is typically easy to detect and identify. When this occurs on an F_Port, the impact is specific to flows involving the F_Port. E_Ports are typically redundant, so severe failures on E_Ports usually only result in a minor drop in bandwidth, as the fabric automatically uses redundant paths.

With moderate cases of faulty media, failures occur, but the port can remain online or transition between online and offline. This can cause repeated errors, which can occur

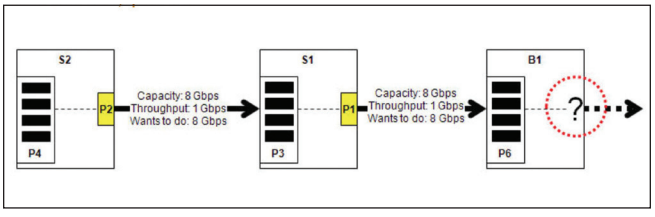

Figure 3: Buffer Congestion
Buffer backup on ingress port 6 on B1 causes congestion upstream on S1, port 3.

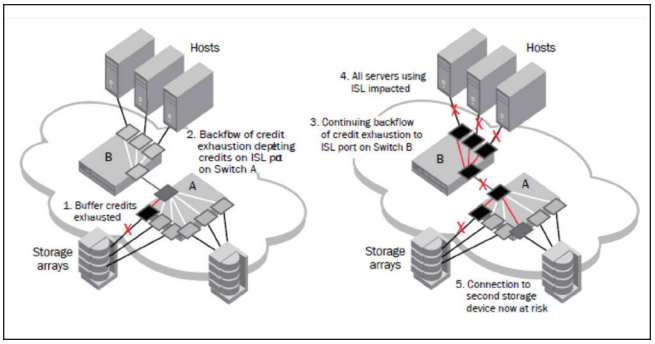

Figure 4: Latency on a Switch Can Propagate Through a Fabric

indefinitely or until the media fails completely. When these types of failures occur on E_Ports, the result can be devastating, as there can be repeated errors that impact many flows. This can result in significant, prolonged impact to applications. Signatures of these types of failures include:

- CRC errors on frames
- Invalid words (includes encoder out errors)
- State changes (ports going offline/online repeatedly)
- Credit loss, the complete loss of buffer credits on an E_Port prevents traffic from flowing, which results in frame loss and I/O failures for devices using the E_Port.

Other reliable indications that an SFP (optic) is likely to fail in the near future include rising SFP temperature, laser power, current draw and voltage.

Automatically Detecting and Mitigating Faulty Media

We discussed Fabric Watch, a SAN software tool, in the previously referenced article. Fabric Watch is used to monitor for CRC errors, invalid words and state changes. You should configure for alerts on reaching low thresholds and fence (disable) a port when reaching high thresholds. Fabric Watch monitors can be enabled to automatically detect most of the faulty media conditions previously noted. For example, Fabric Watch can monitor CRC errors, invalid words and state changes (ports transitioning between offline and online).

Fabric Watch generates alerts based on user-defined thresholds for these conditions. The most common cause of credit loss is corruption to credit return messages (VC_RDY or R_RDY) due to faulty media. Credit corruption is tracked by an encoder out error, which is an invalid word error. Monitoring and mitigating invalid word issues protects against credit loss. Fabric Watch also provides a mechanism that quarantines the badly behaving component with the optional action of port fencing. Port fencing is available for each of the previously noted conditions and is recommended to automatically protect the fabric from these error conditions.

Misbehaving High-Latency Devices

A device experiencing latencies responds more slowly than expected. The device doesn't return buffer credits (through R_RDY primitives) to the transmitting switch fast enough to support the offered load, even though the offered load is less than the maximum physical capacity of the link connected to the device (see Figure 3).

Once it exhausts all available credits, the switch port connected to the device needs to hold additional outbound frames until the device returns a buffer credit. When a device isn't responding in a timely fashion, the transmitting switch must hold frames for longer periods, resulting in high buffer occupancy. This results in the switch lowering the rate at which it returns buffer credits to other transmitting switches. This effect propagates through switches (and potentially multiple switches with devices attempting to send frames to devices attached to the switch with the high-latency device) and ultimately impacts the fabric. The impact to the fabric (and other traffic flows) varies based on the severity of the latency the device exhibits. The longer the delay the device causes in returning credits to the switch, the more severe the problem (see Figure 4).

Severe device latencies result in frame loss, which triggers the host Small Computer System Interface (SCSI) stack to detect failures and retry I/Os. This process can take tens of seconds (possibly as long as 60 seconds), which can cause a noticeable application delay and potentially result in application errors. If the time between successive credit returns by the device is more than 100 milliseconds, then the device is exhibiting severe latency. When a device exhibits severe latency, the switch is forced to hold frames for excessively long periods (hundreds of milliseconds). When the established timeout threshold is exceeded, the switch drops the frame (per FC standards). Frame loss in switches is also known as Class 3 (C3) discards or timeouts.

Since the effect of device latencies often spreads through the fabric, frames can be dropped due to timeouts on the F_Port to which the misbehaving device is connected as well as on E_Ports carrying traffic to the F_Port. Dropped frames typically cause I/O errors that result in a host retry and can result in significant decreases in application performance. The implications of this behavior are exacerbated because frame drops on the affected F_Port (device) result in I/O failures to the misbehaving device (which would be expected), and frame drops on E_Ports may cause I/O failures for unrelated traffic flows involving other hosts (which wouldn't typically be expected).

Once bottleneck detection is enabled, the switch monitors F_Ports for latency symptoms. Specifically, it looks for conditions in which the time delay between successive buffer credit returns from a device is higher than expected. When the condition is detected, reports show latency bottlenecks at F_Ports based on configurable thresholds. These reports can be leveraged to determine the:

- Severity and duration of the latency behavior
- Specific device port on which device latencies are occurring
- Actual device latency in the range of 100 microseconds to hundreds of milliseconds.

Conclusion

Increasingly complex data center architectures require more proactive performance management techniques to ensure application user requirements are met. Nobody wants to go to an Automated Teller Machine (ATM) to withdraw cash and wait 60 seconds; you want your hard-earned cash yesterday. A more resilient SAN fabric goes a long way toward improving application performance and mitigating potential performance problems. **ETJ**

Dr. Steve Guendert is a principal engineer and global solutions architect for Brocade Communications, where he leads the mainframe-related business efforts. A senior member of both the Institute of Electrical and Electronics Engineers (IEEE) and the Association for Computing Machinery (ACM), he serves on the board of directors for the Computer Measurement Group (CMG). He is also a former member of the SHARE Board of Directors.
Email: stephen.guendert@brocade.com
Twitter: @BRCD_DrSteve