MAINFRAME

Brocade FICON/FCP Intermix Best Practices Guide

This guide discusses topics related to mixing FICON and FCP devices in the same Storage Area Network (SAN), focusing on issues that end users need to address. These include specifics for the IBM zEnterprise System, as well as the fabric elements to consider when evaluating a FICON/FCP intermix solution.

BROCADE

CONTENTS

INTRODUCTION

Protocol Intermix Mode (PIM) is a feature supported by IBM on zSeries processors, which allows IBM Fibre Connection (FICON) and open systems Fibre Channel Protocol (FCP) traffic to coexist on the same physical storage network. Over the past decade, Fibre Channel (FC) has become the dominant protocol for connecting servers to storage. It was designed to be a robust, highly reliable, and high-performance foundation layer supporting a framework for both channel and network-oriented upper layers. The result was an ideal architecture for delivering mass connectivity. End users have been able to combine mainframe FICON storage networks and open systems FC SANs onto an enterprise-wide storage network or systems area network since early 2003, when IBM initially announced zSeries support for what is known as "FICON/FCP intermix," or PIM. During that same period, IBM was one of the biggest supporters of Linux in the IT industry, specifically Linux on the mainframe.

Although FICON/FCP intermix has been approved by IBM and FICON director vendors, many end users, particularly large enterprises, have been hesitant to try intermix for several reasons. These reasons include security and management concerns, internal politics, and perhaps a lack of understanding of how FICON/FCP intermix really works. However, considering the cost reductions achieved by FICON/FCP intermix and access to the latest advances in technology—such as the IBM zEnterprise System (z196 and z114), the IBM zEnterprise BladeCenter® Extension (zBX), and N_Port ID Virtualization (NPIV)—FICON/FCP intermix offers a clear advantage for SAN and server network design. This paper discusses the mixing of FICON and FCP devices in the same SAN. It focuses on end-user issues and the fabric elements to consider when evaluating a FICON/FCP intermix solution.

NOTE: In this paper, the term "switch" is used in phrases such as "Fibre Channel switch" to refer to a switch, director, or backbone platform.

INTERMIX CONCEPTS

When the storage networking industry discusses FICON/FCP intermix, the topic discussed is intermix at the connectivity layer, that is, on the same directors, switches, and fiber cable infrastructure. What is not typically discussed is mixing open systems and mainframe disk storage on the same Direct Access Storage Device (DASD) array, a relatively new subject area (that is beyond the scope of this paper).

In open systems environments, Fibre Channel Protocol (FCP) is the upper-layer protocol for transporting Small Computer Systems Interface version 3 (SCSI-3) over Fibre Channel transports. IBM introduced FICON in 1998 to replace the aging Enterprise Systems Connection (ESCON) architecture. Like FCP, FICON is an upper-layer protocol that uses the lower layers of Fibre Channel (FC-0 to FC-3). This common Fibre Channel transport is what enables mainframes and open systems to share a common network and I/O infrastructure, hence the term "intermix."

Mixing FC4 protocols in the same fabric has been discussed since the advent of the Fibre Channel architecture. There are several key concepts to understand when discussing FICON/FCP intermix.

- The FC4 type of an FC frame is an element of the payload; it is not part of the routing algorithms. Therefore, Fibre Channel can function as protocol-agnostic, which allows it to be an ideal common transport technology.
- FICON was the first major FC4 type other than FCP (the SCSI transport protocol in Fibre Channel) to be released in the storage market.
- Other standards-defined FC4 types include IP, IPI (Intelligent Peripheral Interface), and HIPPI (High-Performance Parallel Interface). Of these defined types, FCP is the most widely used at present, with FICON being a close second.

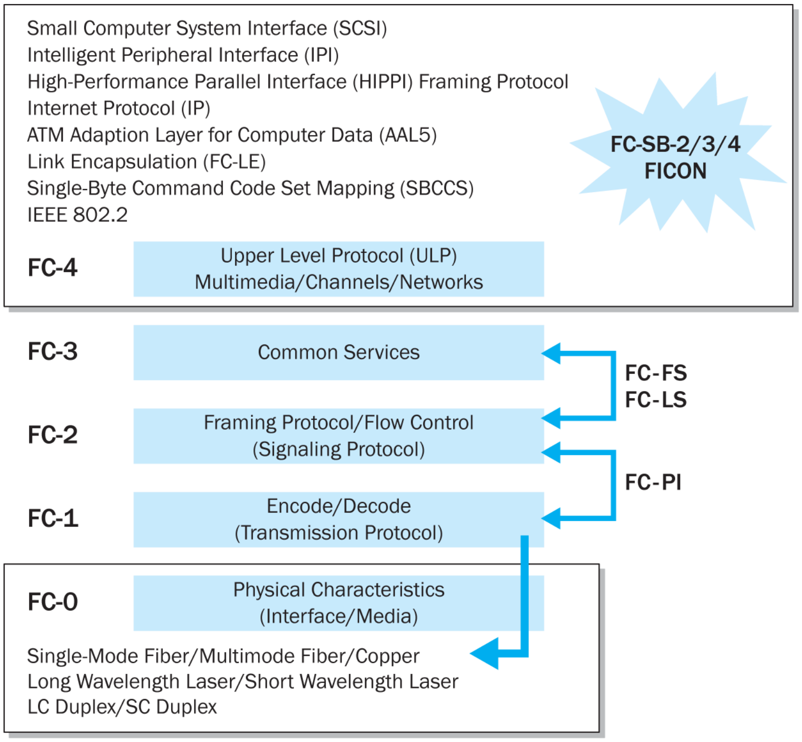
Small Computer System Interface (SCSI)
Intelligent Peripheral Interface (IPI)
High-Performance Parallel Interface (HIPPI) Framing Protocol
Internet Protocol (IP)
ATM Adaption Layer for Computer Data (AAL5)
Link Encapsulation (FC-LE)
Single-Byte Command Code Set Mapping (SBCCS)
IEEE 802.2
FC-SB-2/3/4 FICON
FC-4
Upper Level Protocol (ULP) Multimedia/Channels/Networks
FC-3
Common Services
FC-FS
FC-LS
FC-2
Framing Protocol/Flow Control (Signaling Protocol)
FC-PI
FC-1
Encode/Decode (Transmission Protocol)
FC-0
Physical Characteristics (Interface/Media)
Single-Mode Fiber/Multimode Fiber/Copper
Long Wavelength Laser/Short Wavelength Laser
LC Duplex/SC Duplex

Figure 1. Fibre Channel Structure and Layers

As shown in Figure 1, the open systems SAN Fibre Channel protocol (FC-SCSI-3) and FICON (FC-SB2/3/4) are merely different upper-level protocols in the overall Fibre Channel structure; the difference between the two is at the FC-4 layer (data payload packet). Essentially, an end user's open systems SAN platforms and FICON platforms are identical hardware. However, FICON platforms are typically purchased with a software feature known as Control Unit Port (CUP). The CUP code provides administrators with the capability to manage the FICON directors in-band, using the same host management tools as with ESCON.

WHY INTERMIX FICON AND FCP?

Why would anyone want to intermix both open systems SAN and FICON traffic on the same common storage network? Traditionally, the two environments have not shared system or staff resources, which often operate within completely different cultures. While the concept of PIM may not currently be the implementation strategy for all data centers, the idea is becoming more appealing to many end users. There are five primary reasons to consider moving to an intermix environment:

- Even though native FICON has been available since 2001, many end users still have not migrated to FICON. Data from IBM indicates that new zSeries processors leaving the factory are shipping with significant numbers of ESCON channels per machine. In fact, until IBM released their recent ESCON statements of direction, new processors often had more ESCON channels than FICON channels. On July 12, 2011, IBM made several important new mainframe-related announcements, including this statement of direction: "The IBM zEnterprise 196 (z196) and the IBM zEnterprise 114 (z114) are the last System z servers to support ESCON channels: IBM plans not to offer ESCON channels as an orderable feature on System z servers that follow the z196 (machine type 2817) and z114 (machine type 2818). In addition, ESCON channels cannot be carried forward on an upgrade to such follow-on servers. This plan applies to channel path identifier (CHPID) types CNC, CTC, CVC, and CBY and to features 2323 and 2324. System z customers should continue to eliminate ESCON channels from the mainframe wherever possible. Alternate solutions are available for connectivity to ESCON devices."

Remaining ESCON users will soon start migrating to FICON. Many of those purchasing mainframes with ESCON already have a well-established open systems SAN, which has delivered reliable, high-performance connectivity for several years. Such fabrics, particularly the larger installations, often have available ports on their SAN directors and switches. By considering PIM as a viable option in ESCON-to-FICON migration plans, users might consider a short-term allocation of some unused ports for testing FICON—or even to complete the initial phases of a FICON migration. This may help organizations make their internal technical and financial justifications for a larger FICON migration and let them delay purchase of a separate FICON infrastructure, if they choose not to pursue the PIM option for the production FICON environment.

- It makes good sense to use PIM in specialized non-production environments that require flexibility and resource sharing. Examples are quality assurance, test and development, and even dedicated disaster recovery data center environments. Many FICON users with newer DASD arrays who use FCP for replication and mirroring are already running in PIM.
- The most important reason users move to a PIM environment is to reduce Total Cost of Ownership (TCO). The consolidation of both switching infrastructure and cabling plants when running a PIM environment can significantly reduce TCO. The latest generation of Brocade® high-port-count directors, the Brocade DCX® 8510 and DCX Backbone family, are designed with non-blocking architectures and full throughput to all ports, even when the port count is fully populated. Very large IT organizations with separate open systems and FICON storage networks have installations large enough to fully utilize their switching hardware, even when running distinct, segregated storage networks. However, many smaller organizations also run distinct, segregated storage networks to keep the open systems SAN separate from the FICON/mainframe environment. Sometimes these organizations do not fully utilize their switching hardware; they often have directors or switches with many available ports. Sometimes, up to 50 percent of the director or switch is reserved for "planned growth." In this type of environment, significant savings can be realized by consolidating the underutilized directors onto a common SAN operating in PIM. This results in fewer "serial numbers" on the floor, which in turn leads to lower maintenance, electrical, cooling, and other operational costs. PIM also allows an organization to have one common cable plant/fiber infrastructure, allowing for further reductions in management and operational costs.
- IBM has been one of the biggest proponents of Linux for the past five years—specifically, of consolidation of open systems servers, particularly Wintel servers, onto zLinux partitions. This trend did not really start until the release of the System z9, when NPIV was announced. The trend continued with the System z10 and the zEnterprise 196 (z196), and it is likely to accelerate with the new zEnterprise 114 (z114). The significant enhancements in performance and operational cost savings offered by the z196 and z114 are part of a very compelling zLinux consolidation story. IBM published the results of an internal project in which approximately 3900 open systems servers were consolidated onto mainframes, and the projected TCO savings was 80 percent over five years.
- The zEnterprise BladeCenter Extension (zBX) is an infrastructure component of zEnterprise that houses and supports select IBM BladeCenter blade servers and workload optimizers. It is prebuilt for integration with the System z196 and/or z114. The zBX, when used in conjunction with the zEnterprise Unified Resource Manager (zManager), extends the System z quality of service and robust management capabilities to POWER 7 (AIX) and x86 (Linux) compute elements in the zBX today. With their April 12, 2011 Hardware Announcement (US Hardware Announcement 111-078), IBM announced a statement of direction to support Windows on the zBX: "In the fourth quarter of 2011, IBM intends to offer select IBM System x blades running Microsoft Windows in the IBM zEnterprise BladeCenter Extension Model 002." End users will be able to take even greater advantage of the server consolidation possibilities inherent in the zEnterprise System by also consolidating their I/O infrastructure via implementing PIM for their storage network.

CUSTOMER INTERMIX SOLUTIONS

Here are some representative customer scenarios for using PIM:

- Small FICON and open systems environments with a common storage network
- z/OS servers accessing remote FICON storage via FICON cascading
- Linux on a zSeries mainframe running z/OS, using FICON to access local storage
- Hardware-based remote DASD mirroring between two locations using FCP as a transport, for example, PPRC
- A zSeries processor that accesses storage via FICON and storage devices via FCP to perform remote mirroring between devices (instead of ESCON)
- Open systems servers accessing storage on the same storage network using FCP
- Linux on the zSeries located at either of two sites, using FCP to access storage
- zEnterprise System (z196 or z114) running z/OS workloads (such as DB2) with FICON attached storage, Linux under z/VM accessing SCSI storage via FCP channels, and IBM BladeCenter blade servers in the zBX running AIX, Linux, and (later in 2011) Windows connected to the storage network.

INTEGRATING SYSTEM Z HOSTS AND OPEN SYSTEMS

There are important technical considerations to take into account before making a decision about whether or not to implement PIM. Some of these include fabric management techniques such as zoning, partitioning, binding, and other design considerations.

Integrating System z Hosts and Standalone Open Systems Servers

The primary technical consideration when mixing FCP and FICON arises from the management differences between the two protocols. Therefore, when considering a PIM environment, first consider the management techniques used in each protocol and the management requirements of the target environment.

The mechanism for controlling device communication is the primary management difference. Because FCP and FICON are both FC4 protocols, they do not affect the actual switching of frames. Therefore, the management differences between the two protocols are not relevant, unless you want to control the scope of the switching through zoning or connectivity control. For example, FCP devices use name server zoning as a way to provide *fabric-wide connection control*. FICON devices use the Prohibit Dynamic Connectivity Mask (PDCM, explained in the next section) to provide *switch-wide connection control*.

FICON Connectivity

FICON connectivity is a multilayered function, which is address-centric with regard to the switch ports. FICON communications are address-centric, definition oriented, and use host assignment to determine device communication. The server-to-storage configuration is defined using a host-based program and is stored in the channel drivers. Additional connectivity control is managed at the switch level by configuring (via the PDCM) which port addresses are allowed to form connections with each other. Initially, the FICON architecture was limited to single-switch (domain) environments. Due to the single-byte addressing scheme limitations it inherited from ESCON, the original FICON architecture limited configurations to single domains. At that time, FICON-compliant directors did not form E-ports (Inter-Switch Links, or ISLs) with other switches. If two directors were connected, the FICON Management Server in the director reported this condition as an invalid attachment and prevented the port from coming online. The z/900, z/OS, and accompanying 64-bit architecture announced in mid-2000 allowed for 2-byte addressing. Subsequently, FICON cascading was qualified by IBM and Brocade to provide one-hop configurations between the channel and control units, via ISLs between two switches/directors. Although FICON cascading relieves the single domain restriction, the end user must decide whether or not to exploit FICON cascading functionality.

The FICON architecture communicates port-to-port connectivity via the PDCM that is associated with the port address. The PDCM is a vector-defined addressing system that establishes which addresses are allowed to communicate with each other. The FICON switch is required to support hardware enforcement of the connectivity control, therefore it must be programmed into the port hardware and must apply to all connections. In PIM environments, this creates the potential for the connectivity information to be more restrictive than the zoning information used by open systems devices.

It is also important to understand the implications of FICON port addressing versus port numbering in FCP. Like ESCON, FICON abstracts the concept of the port by creating an object known as the "port address." All of the configuration attributes associated with a port are attached to its port address. A subsequent association is then made between the port address and the port number. The port addressing concept facilitates the ability to perform FICON port swaps for maintenance operations, without the need to regenerate the host configuration. The port address abstraction is not a concept in the Fibre Channel architecture and is therefore foreign to FCP.

FCP Connectivity

In contrast, FCP communications are name-centric, discovery-oriented, fabric-assigned, and use the Fibre Channel Name Server to determine device communication. FCP connectivity is device-centric and defined in the fabric using the World Wide Port Names (WWPNs) of the devices that are allowed to communicate. When an FCP device attaches to the fabric, it queries the Fibre Channel Name Server for the list of devices that it is allowed to form a connection with (that is, the zoning information). FICON devices do not query the Fibre Channel Name Server for accessible devices, because the allowable port/device relationships have been previously defined using the host-based program: IOCP (Input/Output Configuration Program) or HCD (Hardware Configuration Definition). Hence, the zoning and name server information does not need to be retrieved for FICON. In addition, FCP configurations support multiple switch environments (fabrics) and allow as many as seven hops between source (server) and target (storage). FCP port-to-port connectivity has traditionally been enforced via zoning, although other techniques complementary to zoning—such as port, switch, and fabric binding—have been introduced in the past few years. Binding helps alleviate security concerns experienced in PIM installations, because with FCP any device in the fabric can access the ANSI standard FC management server by logging into the fabric.

INTEGRATING SYSTEM Z HOSTS USING Z/OS AND ZLINUX

In addition to the management-centric considerations dicussed above, there are additional issues to consider when installing PIM with z/OS and zLinux. You need to consider some basic capabilities with the z/VM and/or zLinux operating systems, as well as specifics of using FICON channel cards with Linux and NPIV.

The IBM zEnterprise System, like its predecessors, inherits sophisticated virtualization capabilities. Virtualization brings finer control to powerful systems that can be logically divided into multiple simultaneously running processes. The System z provides a hypervisor function, which enables the hosting of multiple logical partitions (LPARs), in which operating system (OS) images can run independently. In addition, z/VM can run as a second-level hypervisor in one or more of these LPARs, thereby creating virtual machines (VMs) in which additional OS images can be run within an LPAR. These significant virtualization capabilities allow concurrent execution of a large number of OS images on a single System z CEC (Central Electronics Complex). This virtualization achieves its most significant economic benefit by sharing and making optimum use of I/O resources.

An IBM zEnterprise System running z/VM and up to 60 LPARs needs a storage network that can a) effectively handle this computing power and b) share the I/O resources in a controlled, secure manner. Predecessors of the IBM zEnterprise System pioneered server virtualization, including the sharing of data storage subsystems among the virtual servers of a host computer using the channel sharing capabilities of FICON channels in Fibre Channel fabrics. This sharing of industry-standard SCSI devices among virtual servers using FCP in SANs, however, has historically been problematic, due to the traditional single-user design and the need to reserve resources for use by particular OS images. Enterprise-class systems, which often host multiple OS images, require each OS image that uses SCSI-FCP protocols to have the presence of an unshared Node Port (N_Port) in a Host Bus Adapter (HBA), in order to be uniquely identified by an N_Port ID.

To apply the power of server virtualization to such an environment, recent IBM mainframes (such as the IBM Systems z196, z114, z10 and z9) implement an FC standard called NPIV, which enables the sharing of host adapters in these IBM mainframes and FC fabrics. With NPIV, a host FC adapter is shared in such a way that each virtual adapter is assigned to a virtual server and is separately identifiable in the fabric. Connectivity and access privileges in the fabric are controlled via the identification of each virtual adapter. The Systems z196, z114, z10 and z9 use NPIV and FC virtual switches in virtual fabrics to support Linux on System z. Storage networks are now expanding into virtual realms, which have been widely accepted in mainframe environments for decades. Mainframes have housed many virtual processes to increase performance, manageability, and utilization. The same transformation is occurring in storage networks on two levels. First, NPIV lets Linux images on System z servers access open systems storage. This storage usually costs much less than the high-end DASD associated with mainframes. Secondly, virtual fabric technology lets the storage network create multiple FC virtual switches using the same physical hardware.

Many open systems servers have appeared in the data center to fulfill different business application requirements. Every time a department finds a new application, servers are acquired that require new switching, storage, cabling, and management. These open systems servers introduce additional administrative overhead such as maintenance, procurement, and inventory costs. The acquisition of a few servers at a time over several years can result in racks and racks of servers, many of which are underutilized. However, instead of acquiring new hardware for new or overloaded applications, you can replace these open systems servers with virtual Linux on System z servers. These virtual servers can be up and running quickly, instead of it taking days or weeks to acquire and install physical open systems servers. Implementing Linux on System z servers means that world-class computing resources can now take advantage of low-cost open systems storage, further reducing costs. In addition, System z servers running Linux scale predictably and quickly and offer benefits such as consistent, homogeneous resource management.

FCP Channels on the Mainframe

The IBM System z FICON Express2, FICON Express4, FICON Express8, and FICON Express8S channel cards provide support for FC and SCSI devices in Linux environments. The I/O component that accesses a SAN using the SCSI-FCP protocol is called an FCP channel. An FCP channel provides the same FC adapter functionality as that typically provided by Peripheral Component Interconnect (PCI)-based HBAs on other platforms. FCP channels also contain hardware and firmware to support protocol offload, enhanced Reliability, Availability, and Serviceability (RAS), and—most importantly—shared access by multiple OS images running in various LPARs, VMs, or both on the mainframe. The FCP channel feature is available with all FICON Express adapters (FICON Express, FICON Express2, FICON Express4, FICON Express8, and FICON Express8S).

These channel cards provide either FICON or FCP channel functionality, depending on the firmware/microcode with which they are loaded. The channel card itself features local memory, a high-speed Direct Memory Access (DMA) engine used for accessing system memory, an FC HBA chipset, and a pair of cross-checked PowerPC processors. The firmware itself comprises five major components:

- A device driver that interfaces directly with the FC HBA chipset to drive the FC link
- A real-time OS kernel that provides the basic system infrastructure
- A layer unique to the SCSI protocol for providing the programming interface to the host device drivers for SCSI devices

- A channel and control unit component providing the interface to the z/Architecture I/O subsystem, which in turn provides I/O and other operations used in controlling the configuration
- A Queued Direct I/O (QDIO) component for providing the primary transport mechanism for all FC traffic

FCP support on today's mainframes allows Linux running on the host to access industry-standard SCSI storage devices. For disk applications, these FCP storage devices utilize Fixed Block Architecture (FBA) 512-byte sectors, rather than Extended Count Key Data (ECKD) format. The SCSI and FCP controllers and devices can be accessed by Linux on the System z as long as the appropriate I/O driver support is present. There are currently two supported methods of running Linux on System z: natively in a logical partition or as a guest operating system under z/VM (version 4 release 3 and later releases).

The FCP I/O architecture used by System z and zSeries mainframes fully complies with the Fibre Channel Standards specified by the InterNational Committee of Information Technology Standards (INCITS). To review: FCP is an upper-layer FC mapping of SCSI on a common stack of FC physical and logical communications layers. FC-FCP and SCSI are supported by a wide range of controllers and devices, which complement the storage attachment capability through FICON and ESCON channels.

The QDIO architecture is used by FICON channels in FCP mode to communicate with the operating system. This architecture is derived from the same QDIO architecture defined for HiperSockets communications and for OSA Express. Rather than using control devices, FCP channels use data devices that represent QDIO queue pairs, which consist of a request queue and a response queue. Each of these queue pairs represents a communications path between the FCP channel and the operating system. An operating system can send FCP requests to the FCP channel via the request queue, and the response queue can be used by the FCP channel for passing completion indications and unsolicited status indications back to the operating system.

The FCP channel type still needs to be defined using HCD/IOCP, as do the QDIO data devices. However, there is no requirement for defining the FC storage devices or switches. All of these devices are configured on an operating system level using the parameters outlined in the industry-standard FC architecture. They are addressed using World Wide Names (WWNs), FC Identifiers (IDs), and Logical Unit Numbers (LUNs). After the addresses are configured on the operating system, they are passed to the FCP channel, along with the corresponding I/O request, via a queue.

LUN Access Control and NPIV

The FCP industry-standard architecture does not exploit the security and data access control functions of the mainframe Multiple Image Facility (MIF). Prior to System z9 and NPIV, the z990 and z890 had a feature known as FCP LUN access control. LUN access control provides host-based control of access to storage controllers and their devices, as identified by LUNs. LUN access control also allows for read-only sharing of FCP SCSI devices among multiple OS images. When a host channel is shared among multiple OS images, the access control mechanism is capable of providing for either none or all of the images to have access to a particular logical unit (device) or storage controller. FCP LUN access control provides the ability to define individual access rights to storage controller ports and devices for each OS image. LUN access control can significantly reduce the number of FCP channels needed to provide controlled access to data on FCP SCSI devices. Without LUN access control, FCP channels prevent logical units from being opened by multiple Linux images at the same time. In other words, access is granted on a first-come, first-served basis. This prevents problems with concurrent access from Linux images that are sharing the same FCP channel (also sharing the same WWPN). This means that one Linux image can block other Linux images from accessing the data on one or more logical control units. If the FCP channel is used by multiple independent operating systems (under z/VM or in multiple LPARs), the SAN is not aware of this fact, and it cannot distinguish among the multiple independent users. You can partially avoid this by requiring separate FCP channels for each LPAR. However, this does not solve the problem with z/VM.

Another and better method to control access to devices is the implementation of NPIV. In the mainframe space, NPIV is unique to the System z9 and later IBM System z machines. NPIV allows each operating system sharing an FCP channel to be assigned a unique virtual WWPN. The virtual WWPN can be used for both device-level access control in a storage controller (LUN masking) and for switch-level access control on a Fibre Channel switch (via zoning). In summary, NPIV allows a single physical FCP channel to be assigned multiple WWPNs and appear as multiple channels to the external SAN environment. These virtualized FC N_Port IDs allow a physical FC port to appear as multiple distinct ports, providing separate port identification and security within the fabric for each OS image. The I/O transactions of each OS image are separately identified, managed, transmitted, and processed as if each OS image had its own unique physical N_Port.

NPIV is based on extensions made to the Fibre Channel standards that allow an HBA to perform multiple logins to the SAN fabric via a single physical port. The switch to which the FCP channel is directly connected (that is, the “entry switch”) must support multiple N_Port logins. No changes are required for the downstream switches, devices, or control units. The switch—and not the FCP channel—provides the multiple WWPNs used for the virtualization. Therefore, NPIV is not supported with FCP point-to-point attachments. Rather, NPIV requires switched FCP attachment. NPIV is available with Linux on System z in a LPAR or as a guest of z/VM versions 4.4, 5.1, and later for SCSI disks accessed via dedicated subchannels and for guest IPLs. For guest use of NPIV, z/VM versions 4.4, 5.1, 5.2, and later all provide support transparently (no Program Temporary Fix [PTF] is required). z/VM 5.1 and later provide NPIV support for VM system use of SCSI disks (including emulated FBA minidisks for guests). z/VM 5.1 requires a PTF to properly handle the case of a Fibre Channel switch not being able to assign a new N_Port ID when one is requested (due to the switch’s capacity being exceeded).

In summary, while LPARs consume one or more FC ports, NPIV allows multiple zLinux “servers” to share a single FC port. The sharing allows you to maximize asset utilization, because most open systems applications require little I/O. While the actual data rate varies considerably for open systems servers, a rule of thumb is that they consume about 10 MB/sec. Fibre Channel has reduced the I/O latency by scaling FC speeds from 1 Gbps to 2 Gbps, to 4 Gbps, to 8 Gbps, and now to 16 Gbps. With each Gigabit per second (Gbps) of bandwidth, one FC port supplies 100 MB/sec (Megabyte per second) of throughput. A 4-Gbps link should be able to support about 40 zLinux severs, from a bandwidth perspective. To aggregate many zLinux servers on a single FC port, the Fibre Channel industry has standardized NPIV, which lets each zLinux server (on a System z machine) have its own 3-byte FC address, or N_Port ID. After N_Ports (servers or storage ports) have acquired an N_Port ID from logging into the switch, NPIV lets the port request additional N_Port IDs—one for each Linux on Systems z servers running z/VM version 5.1 or later. With a simple request, the switch grants a new N_Port ID and associates it to the Linux on System z image. The WWN uniquely identifies the Linux on System z image and allows for the zLinux servers to be zoned to particular storage ports.

VIRTUAL FABRICS

Considering that a System z196 can have up to 288 FICON Express 8S (8-Gbps) ports, thousands of N_Port IDs can be quickly assigned to a mainframe. Managing this many ports in a single fabric can become cumbersome and cause interference. To isolate applications running behind different N_Ports attached to the same switch, the T11 Fibre Channel Interfaces technical committee (www.t11.org) has standardized virtual fabrics. Similar to the way the IBM zEnterprise System has multiple virtual servers, a physical chassis can support up to 4,095 Fibre Channel switches.

Fibre Channel virtual switches relieve another difficulty (as shown in Figure 2), which is managing several distributed fabrics. The storage network has grown organically with different applications and the physical limitations of the switches. Table 1 shows the port counts for each fabric on each site. Fabrics 1 and 2 each have 64-port directors with 28–48 of these ports being in use. The backup fabric has only 24-port switches, and only a single port is available on each of these switches. While Fabrics 1 and 2 have a considerable number of unused ports, the switches do not have the capability to offer ports to the backup fabric. The usage rates of the fabrics are also relatively low, and the combinations of products, firmware releases, and management applications makes the distributed fabric rather complex.

The benefits of the consolidated approach include efficiency, reduced cost, and manageability. Figure 3 illustrates a better solution that meets all of the needs of the data center. The physical configuration of the data center can be consolidated on two 256-port directors, which can be divided into multiple virtual storage networks. These networks can be large or small, and they offer independent, intelligible management.

When large corporations require more than a thousand servers in their data center, the manageability and accountability for controlling storage networks can become unruly. Breaking the fabrics into small virtual fabrics increases the manageability of the storage network. Instead of coordinating a team of administrators to manage one large storage network, individuals each handle a manageable piece of the solution. The virtual nature of the new data center creates a management hierarchy for the storage network and enables administration to be accomplished in parallel.

A further explanation of the differences between the distribute storage network and the consolidated storage network illuminates the benefits of the new approach. In Figure 2, the distributed storage network features 58 open systems servers, consuming 58 FC ports and the corresponding cabling and rack space. Each open systems server is using only a fraction of the link's bandwidth as the speed of the link increases to 4 Gbps. The reliability of the variety of servers that have accumulated over the years is significantly less than the tested reliability of the System z196. The administrative cost of adding and repairing servers that change every quarter leads to complexity and inefficiency. In a nutshell, the organic growth of the open systems servers has led to a population problem.

The consolidated data center in Figure 3 features a unified architecture that is adaptable and efficient. The System z196 supports mixed applications and is managed from either a single screen or multiple screens. The director has been divided up into Fibre Channel virtual switches that suit the needs of each application and, in the future, will be able to quickly scale to almost any level. The links in Figure 3 are black because they can be attached to any of the virtual fabrics. The combination of more powerful hardware and virtual processors and switches creates a simpler architecture than the distributed storage network.

Figure 4 shows details of the virtualization techniques between the System z196 and the director. The mainframe applications have traditional direct links to the Fibre Channel virtual switches, which connect the mainframe to the DASD. These links typically run at 800 MB/sec and can support high I/O per second (IOPS). The open systems applications are using NPIV to allow them to access the open systems storage. Figure 4 shows how one physical link to Virtual Fabric 1 supports eight open systems applications. Each open systems application has a zLinux server, WWN, and N_Port ID, for management purposes. With multiple open systems applications using a single link, the usage rates have increased to the levels shown in Table 2. Table 2 also shows how fewer ports were used more efficiently in the consolidated storage network than in the distributed storage network. Even after several applications were added to the data center, the number of ports in the fabric decreased from 199 to 127.

With the link speed increasing to 8 Gbps, the utilization rate has increased by more than three times, to an average of 61 percent from 20 percent. With virtual fabrics, the number of ports used in each fabric is flexible, and switch ports can be added to the fabric without the addition of another switch. Other virtual switches also can be present in the director, which is a configuration with a higher reliability than multiple small switches.

PIM and NPIV

PIM via NPIV and virtual fabrics provides near-term solutions such as cloud computing and computing on demand. To support automated computing power in these environments, the processors, storage, and storage networking must be driven by policy-based applications. The administrator establishes the performance policies and enables the soft, or virtual, layers on top of the hardware to automatically manipulate resources to meet data center demands. PIM enables consolidation, cost savings, a "greener" solution, and ease of management, as shown in Figures 2, 3, and 4 and Tables 1 and 2.

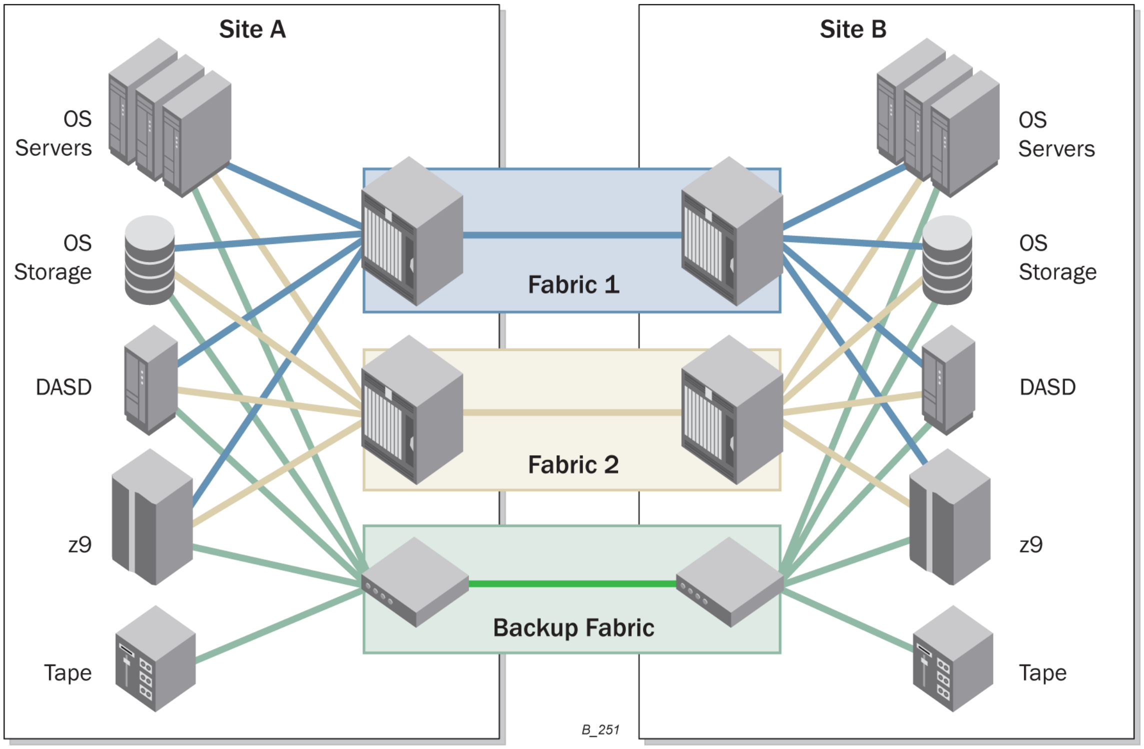

Figure 2. Dispersed Storage Network

Table 1. Distributed Storage Network Port Count and Utilization Rate

	Fabric	Port Count	Ports Used	Link Speed (Gbps FC)	Average Throughput (MB/sec)	Utilization Rate (C/O)
Site A Switch	1	64	48	1	42	42
	2	64	34	2	36	18
	Backup	24	23	2	28	14
Site B Switch	1	64	43	1	25	25
	2	64	28	2	36	18
	Backup	24	23	2	18	9
TOTAL			199		30	20

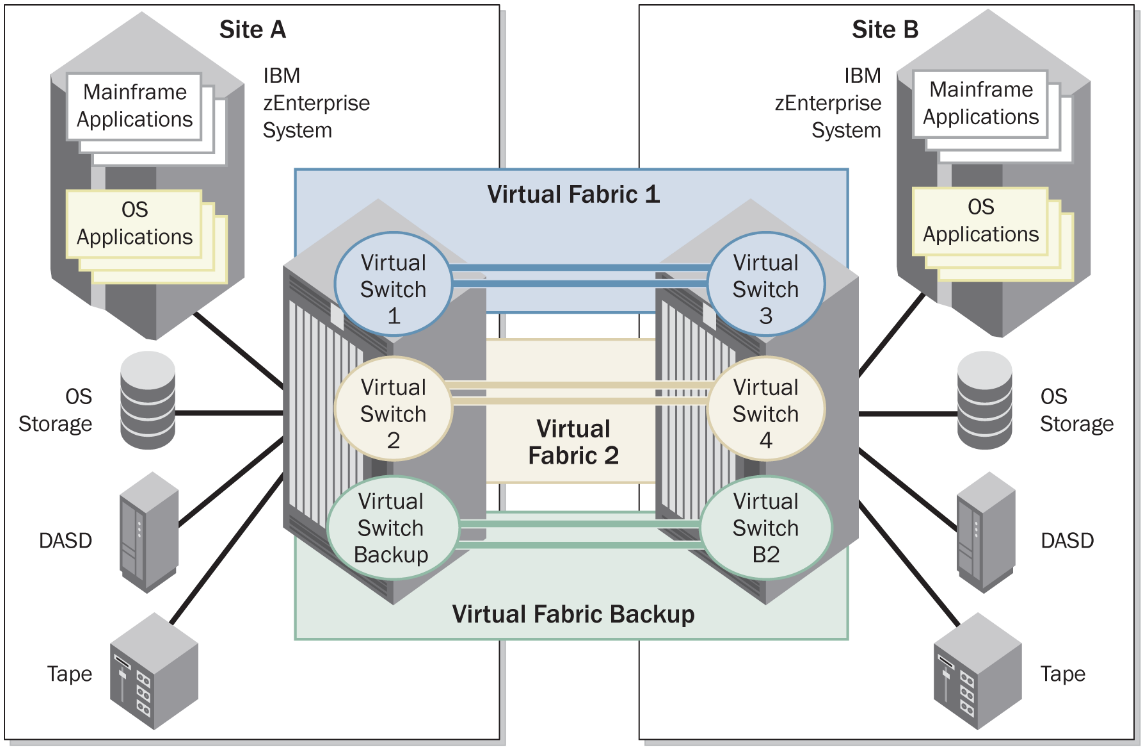

Figure 3. Consolidated Storage Network

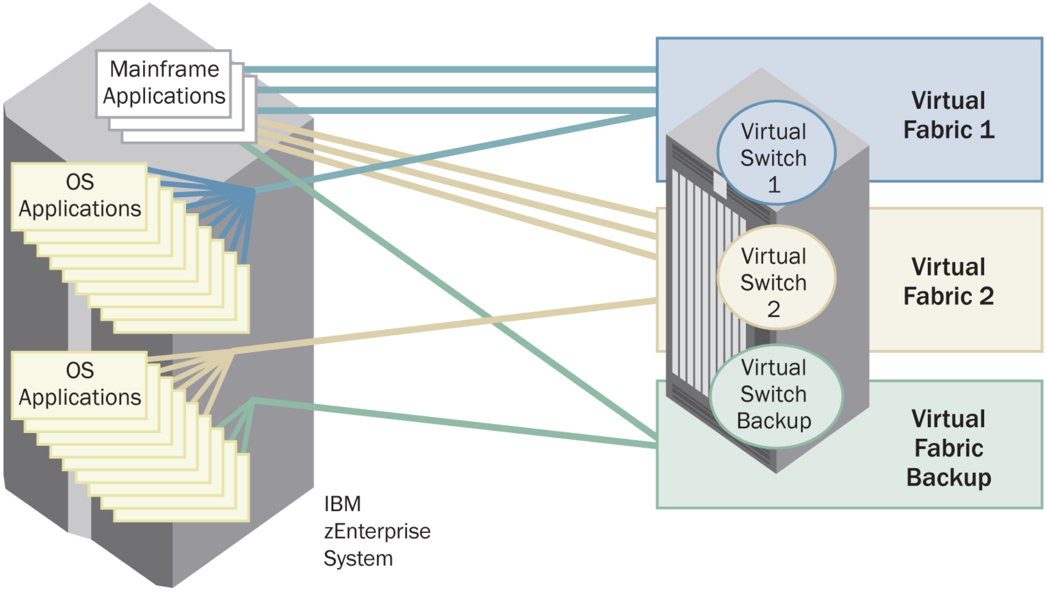

Figure 4. Virtualized Storage Network

Table 2. Consolidated Storage Network Port Count and Utilization Rate

	Fabric	Port Count	Ports Used	Link Speed (Gbps FC)	Average Throughput (MB/sec)	Utilization Rate (C/O)
Site A Switch **256-Port Director**	1	Up to 256	26	8	576	72
	2	Up to 256	18	8	552	56
	Backup	Up to 256	16	8	332	41
Site B Switch **256-Port Director**	1	Up to 256	28	8	626	78
	2	Up to 256	22	8	530	66
	Backup	Up to 256	17	8	312	39
TOTAL			127		494	61

CONCLUSION

The enhancements made to the IBM System z family have made FICON/FCP intermix, or PIM, a more attractive, viable, and realistic option for connectivity. NPIV technology finally makes it realistic to run open systems and z/OS on a mainframe that connects everything onto a common storage network, using FICON Express channel cards. Virtual fabric technologies, coupled with the IBM zEnterprise System, allow for the end user to bring UNIX and Windows servers into this storage network.

Brocade has worked with IBM and other companies to help develop the T11 standards to support these storage networking virtualization techniques on System z9 and newer IBM mainframes, in order to replace multiple open systems servers with Linux on System z. NPIV lets these servers use open systems storage for a low-cost solution with enhanced technologies. The Fibre Channel virtual switches that create virtual fabrics, such as those on the Brocade DCX Backbone family, offer an additional capability to increase manageability and enable physical switches to be more adaptable.

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.