EMC²

Networked Storage Concepts and Protocols

Version 3.0

- Designing a SAN
- FC SAN Concepts
- IP SAN Concepts

Mark Lippitt
Erik Smith

Part number H4331.5

Contents

Chapter 3 FC SAN Concepts

Figures

Preface

This EMC Engineering TechBook provides fundamental information about Fibre Channel. It presents fabric design considerations, explains how SAN technology works, and describes IP SAN concepts. Extended distance technologies and solutions are also discussed.

For more information on extended distance technologies, refer to the Extended Distance Technologies TechBook, available on the E-Lab Interoperability Navigator (ELN), at http://elabnavigator.EMC.com.

E-Lab would like to thank all the contributors to this document, including EMC engineers, EMC field personnel, and partners. Your contributions are invaluable.

As part of an effort to improve and enhance the performance and capabilities of its product lines, EMC periodically releases revisions of its hardware and software. Therefore, some functions described in this document may not be supported by all versions of the software or hardware currently in use. For the most up-to-date information on product features, refer to your product release notes. If a product does not function properly or does not function as described in this document, please contact your EMC representative.

Note: This document was accurate at publication time. New versions of this document might be released on EMC Online Support at https://support.EMC.com. Check to ensure that you are using the latest version of this document.

Audience

This TechBook is intended for EMC field personnel, including technology consultants, and for the storage architect, administrator, and operator involved in acquiring, managing, operating, or designing a networked storage environment that contains EMC and host devices.

EMC Support Matrix and E-Lab Interoperability Navigator

For the most up-to-date information, always consult the *EMC Support Matrix* (ESM), available on the E-Lab Interoperability Navigator (ELN), at http://elabnavigator.EMC.com.

Related documentation

Related documents include:

- The following documents, including this one, are available through the E-Lab Interoperability Navigator, **Topology Resource Center** tab, at http://elabnavigator.EMC.com.

  These documents are also available at the following location:

`http://www.emc.com/products/interoperability/topology-resource-center.htm`

  - *Backup and Recovery in a SAN TechBook*
  - *Building Secure SANs TechBook*
  - *Extended Distance Technologies TechBook*
  - *Fibre Channel over Ethernet (FCoE): Data Center Bridging (DCB) Concepts and Protocols TechBook*
  - *Fibre Channel SAN Topologies TechBook*
  - *iSCSI SAN Topologies TechBook*
  - *Networking for Storage Virtualization and RecoverPoint TechBook*
  - *WAN Optimization Controller Technologies TechBook*
  - *EMC Connectrix Products Data Reference Manual*
  - *Legacy Information Reference Manual*
  - *Non-EMC Products Data Reference Manual*

- *EMC Support Matrix*, available through E-Lab Interoperability Navigator at http://elabnavigator.EMC.com > **PDFs and Guides**
- RSA security solutions documentation, which can be found at http://RSA.com > **Content Library**

All of the following documentation and release notes can be found on the EMC Online Support site at https://support.EMC.com.

EMC hardware documents and release notes include those on:

- Connectrix B series
- Connectrix M series
- Connectrix MDS (release notes only)
- VNX series
- CLARiiON
- Celerra
- Symmetrix

EMC software documents include those on:

- ControlCenter
- RecoverPoint
- Invista
- TimeFinder
- PowerPath

The following E-Lab documentation is also available:

- Host Connectivity Guides
- HBA Guides

For Cisco and Brocade documentation, refer to the vendor's website.

- http://cisco.com
- http://brocade.com

Authors of this TechBook

This TechBook was authored by Mark Lippitt and Erik Smith, with contributions from the following EMC employees: Kieran Desmond, Ger Halligan, and Ron Stern, along with other EMC engineers, EMC field personnel, and partners.

Mark Lippitt is a Technical Director in EMC E-Lab with over 30 years experience in the storage industry, including Engineering and Marketing roles at Data General, Tandem Computers, and EMC. Mark initiated and led the Stampede project in 1997, which became EMC's first Connectrix offering. Mark is an active T11 participant, a committee within the InterNational Committee for Information Technology Standards, responsible for Fibre Channel Interfaces.

Erik Smith is a Consulting Technologist for the Connectrix business unit within EMC Engineering. Over the past 14 years, Erik has held various technical roles in both EMC Engineering and Technical Support. Erik has authored and coauthored several EMC TechBooks. Erik is also a member of T11.

Conventions used in this document

EMC uses the following conventions for special notices:

IMPORTANT

An important notice contains information essential to software or hardware operation.

Note: A note presents information that is important, but not hazard-related.

Typographical conventions

EMC uses the following type style conventions in this document.

Normal	Used in running (nonprocedural) text for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • Names of resources, attributes, pools, Boolean expressions, buttons, DQL statements, keywords, clauses, environment variables, functions, and utilities • URLs, pathnames, filenames, directory names, computer names, links, groups, service keys, file systems, and notifications
Bold	Used in running (nonprocedural) text for names of commands, daemons, options, programs, processes, services, applications, utilities, kernels, notifications, system calls, and man pages Used in procedures for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • What the user specifically selects, clicks, presses, or types
Italic	Used in all text (including procedures) for: • Full titles of publications referenced in text • Emphasis, for example, a new term • Variables
`Courier`	Used for: • System output, such as an error message or script • URLs, complete paths, filenames, prompts, and syntax when shown outside of running text
`Courier bold`	Used for specific user input, such as commands
`Courier italic`	Used in procedures for: • Variables on the command line • User input variables

	Used in procedures for: • Names of interface elements, such as names of windows, dialog boxes, buttons, fields, and menus • What the user specifically selects, clicks, presses, or types
Italic	Used in all text (including procedures) for: • Full titles of publications referenced in text • Emphasis, for example, a new term • Variables
`Courier`	Used for: • System output, such as an error message or script • URLs, complete paths, filenames, prompts, and syntax when shown outside of running text
`Courier bold`	Used for specific user input, such as commands
`Courier italic`	Used in procedures for: • Variables on the command line • User input variables

Where to get help

EMC support, product, and licensing information can be obtained as follows.

EMC support, product, and licensing information can be obtained on the EMC Online Support site as described next.

Note: To open a service request through the EMC Online Support site, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or to answer any questions about your account.

Product information

For documentation, release notes, software updates, or for information about EMC products, licensing, and service, go to the EMC Online Support site (registration required) at:

https://support.EMC.com

Technical support

EMC offers a variety of support options.

Support by Product — EMC offers consolidated, product-specific information on the Web at:

https://support.EMC.com/products

The Support by Product web pages offer quick links to Documentation, White Papers, Advisories (such as frequently used Knowledgebase articles), and Downloads, as well as more dynamic content, such as presentations, discussion, relevant Customer Support Forum entries, and a link to EMC Live Chat.

EMC Live Chat — Open a Chat or instant message session with an EMC Support Engineer.

eLicensing support

To activate your entitlements and obtain your Symmetrix license files, visit the Service Center on https://support.EMC.com, as directed on your License Authorization Code (LAC) letter e-mailed to you.

For help with missing or incorrect entitlements after activation (that is, expected functionality remains unavailable because it is not licensed), contact your EMC Account Representative or Authorized Reseller.

For help with any errors applying license files through Solutions Enabler, contact the EMC Customer Support Center.

If you are missing a LAC letter, or require further instructions on activating your licenses through the Online Support site, contact EMC's worldwide Licensing team at licensing@emc.com or call:

- North America, Latin America, APJK, Australia, New Zealand: SVC4EMC (800-782-4362) and follow the voice prompts.
- EMEA: +353 (0) 21 4879862 and follow the voice prompts.

We'd like to hear from you!

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to: techpubcomments@emc.com

Your feedback on our TechBooks is important to us! We want our books to be as helpful and relevant as possible. Send us your comments, opinions, and thoughts on this or any other TechBook to:

TechBooks@emc.com

1 Introduction

This chapter describes the differences between channels, networks, Fibre Channel fabric with network characteristics, and SANs, and discusses the differences in physical and logical topologies.

Channels versus networks versus SANs

This section describes the differences between channels, networks, and Fibre Channel fabrics, providing a starting point for the discussions to follow in this document.

Channels

In the past, host computer operating systems have communicated with storage devices over channel connections, such as parallel bus and tag, ESCON, and SCSI. These channel technologies provide fixed connections between host systems and their peripheral devices.

When using channels, static connections are defined to the operating system in advance. Tight integration between the transmission protocol and the physical interface minimizes the overhead required to establish communication and transport large amounts of data to the statically defined devices.

Some characteristics of channel technologies are:

- High performance
- Low protocol overhead
- Static configuration
- Short distance (although ESCON and FICON are exceptions to this rule)
- Connectivity within a single system

Networks

Network technologies are more flexible than channel technologies, and provide greater distance capabilities. Most networks provide connectivity between client or host systems, and carry a variety of data between the devices. A simple example is a network of desktop PCs within a company. This type of setup can provide each PC with connectivity to file and print services, server-based applications, and corporate intranets.

The networks these PCs are connected to provide shared bandwidth and the ability to communicate with many different systems. This flexibility results in greater protocol overhead and reduced performance.

Some characteristics of network technologies are:

- Lower performance than a channel
- Higher protocol overhead
- Dynamic configuration
- Long distance
- Connectivity among different systems.

Fibre Channel: Channels with network characteristics

Fibre Channel captures some of the benefits of both channels and networks. A Fibre Channel fabric is a switched network, providing a set of generic, low-level services onto which host channel architectures and network architectures can be mapped. It provides a serial data transfer interface that operates over copper wire and/or optical fiber at data rates currently up to 1600 MB/s. Networking and I/O protocols (such as SCSI commands) are mapped to Fibre Channel constructs and then encapsulated and transported within Fibre Channel frames. This process allows high-speed transfer of multiple protocols over the same physical interface. SCSI and Fibre Channel

Before designing your fabric, it is essential to understand why Fibre Channel is so widely used and accepted in the data center and why it is the natural technological progression from direct-attach SCSI devices. The phrase *Fibre Channel* is often used as an abbreviation of *SCSI over Fibre Channel*. Fibre Channel is a transport protocol that allows mapping other service-oriented or device-oriented protocols within its transport frames. SCSI over Fibre Channel allows us to overcome the distance, dynamic flexibility, and accessibility limitations associated with traditional direct-attach SCSI.

As with direct-attach SCSI, Fibre Channel provides the block level access to the devices that allows the host system to identify the device as a native device. The true power of native device identification is seen in our ability to use all of our current applications (for example: backup software, volume management, and raw disk management) without modification. With this understanding, the considerations and methodologies needed to build Fibre Channel SANs will be discussed.

Storage Area Networks (SANs)

Figure 1 shows an example of a SAN.

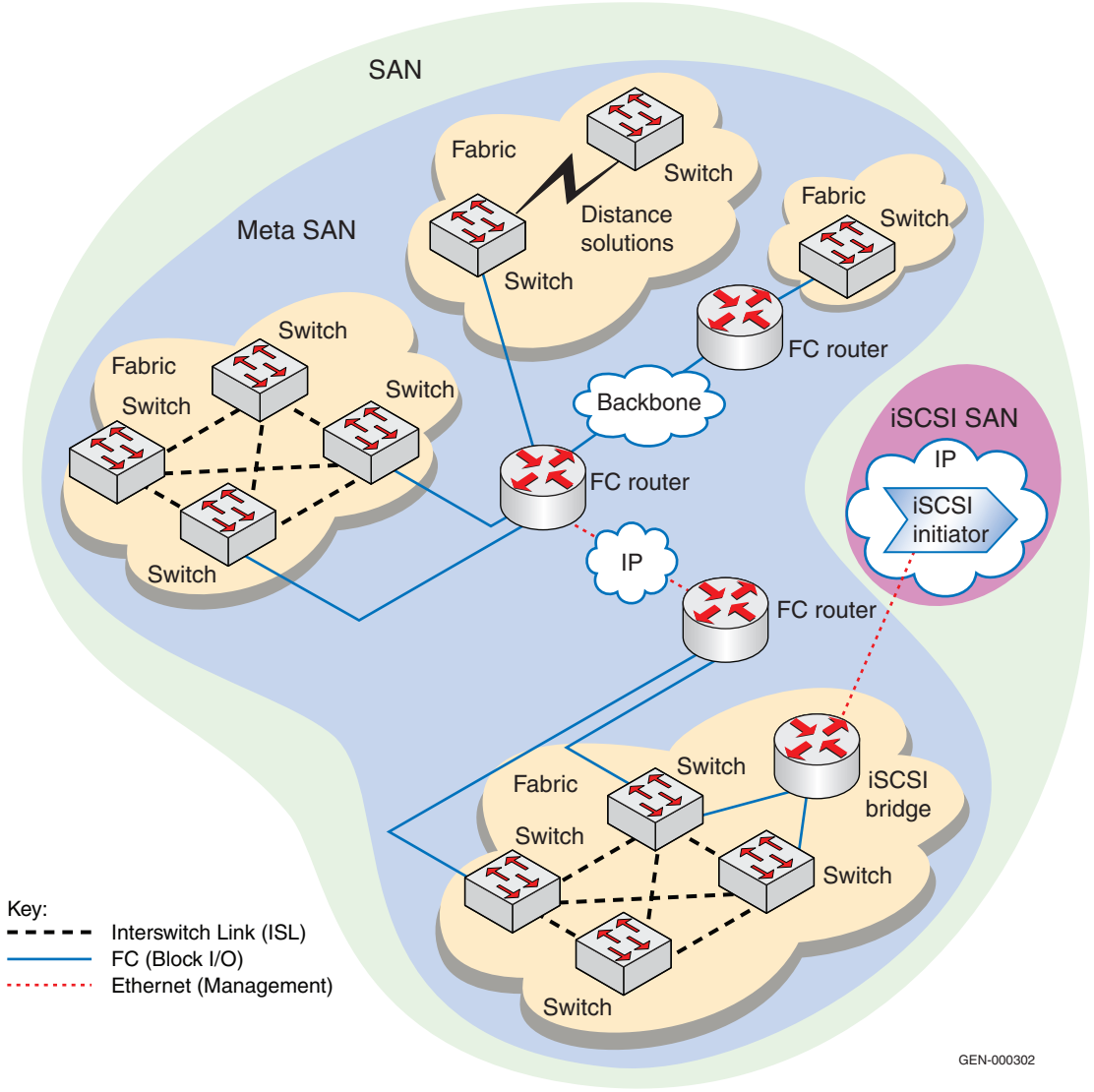


Figure 1 SAN example

Note: Although the FC initiators and FC targets have been omitted from the diagram, it should be understood that the graphic is intended to highlight the majority of supported end-to-end connectivity configurations.

A Storage Area Network is a network that has been created to facilitate block, file, or object I/O operations. SANs can be further broken down into three subcategories, each further discussed in this section:

- "FC SANs," next
- "iSCSI SANs" on page 25
- "Multiprotocol SANs" on page 25

FC SANs

Fibre Channel Storage Area Networks exclusively use Fibre Channel for the transport of command, data, or status information. Fibre Channel SANs consist of either an individual fabric or multiple fabrics interconnected using a routing function (Figure 1 on page 22). As can be seen in Figure 1, FC SANs can be further broken down into fabrics and Meta SANs. A Meta SAN could also be considered to be an iSAN, (refer to "Meta SAN" on page 25), depending on how the routing function is accomplished.

Fabric

A *fabric* is a collection of Fibre Channel ports that all share the same 24-bit address space. Information about ports that are logged into a particular fabric is distributed to all switching elements in the fabric. When fabrics were first introduced, every physical chassis connected by an ISL could be considered to be in the same fabric. Today, with the introduction of VSANs, virtual switches, and fabric routing, this physical relationship breaks down and the VSAN, virtual switch ID, or Fabric ID needs to be considered when trying to determine which ports are in the same fabric. For more information on VSANs, virtual switches, or Fabric routing, see "Virtual switches" on page 324, "VSANs" on page 325, and "Fibre Channel Routing" on page 300.

Although it is not a best practice, a fabric can span long distances by using specialized distance extension equipment or by using a protocol such as FCIP (refer to "FCIP" on page 24). Some may argue that the use of FCIP in a fabric makes it an IP SAN, but this is not the case. A fabric that spans a distance by using an FC-over-IP link still shares the same address space. An argument can be made that such a fabric is a multiprotocol SAN, as described in "Multiprotocol SANs.")

FCIP

FCIP (Fibre Channel over IP), defined in RFC (Request for Comments[1]) 3821, is an IP-based storage networking technology developed by the Internet Engineering Task Force. RFC 3821 describes mechanisms that allow the interconnection of islands of Fibre Channel SANs over IP networks to form a unified SAN in a single Fibre Channel fabric. The motivation behind defining these interconnection mechanisms is a desire to connect physically remote FC sites allowing remote disk access, tape backup, and live mirroring.

FCIP enables the transmission of information by tunneling data between storage area network (SAN) facilities over IP networks. The tunneling functionality is achieved by encapsulating Fibre Channel frames inside of TCP/IP datagrams as defined in RFC 3643 and FC-BB-2.

As shown in Figure 2, depending on the MTU size, a FC frame can be a part of one or more TCP segments.

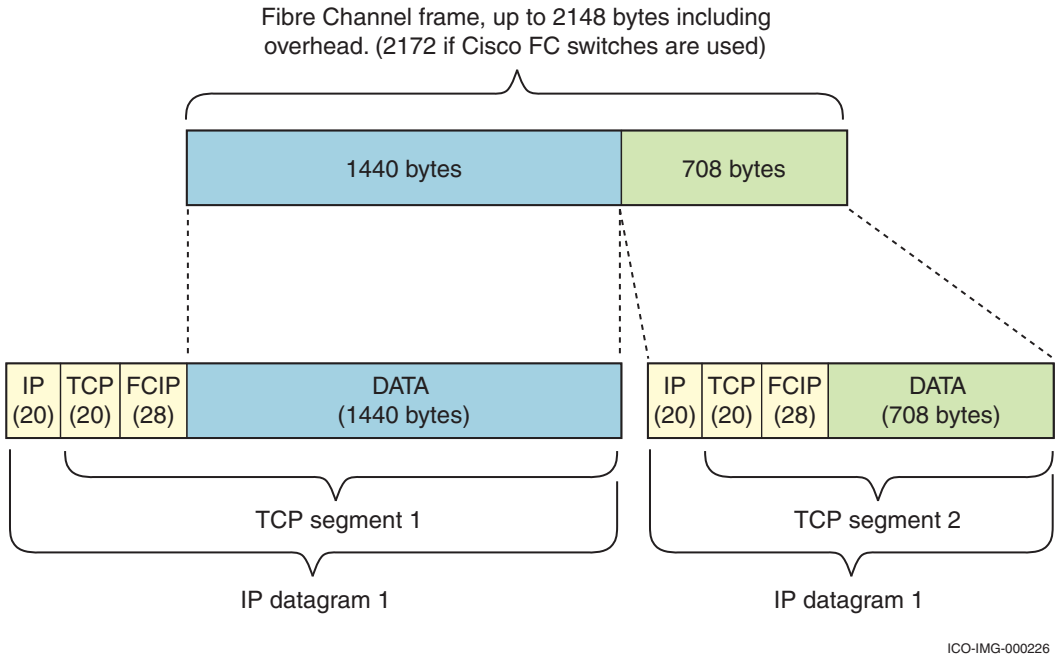


Figure 2 **FCIP-MTU 1500**

1. Request for Comments (RFC) are official documents from the Internet Engineering Task Force (iETF) with unlimited distribution. These are numbered in a series and are referred to by numbers.

FCIP is designed to be transparent to Fibre Channel. In most cases, EMC customers would utilize FCIP technology in one of two ways:

- To allow EMC SRDF® or MirrorView™ operations to be conducted across an IP network.
- To connect Fibre Channel SAN *islands*, essentially extending the reach of the fabric.

E-Lab Navigator describes the qualified devices and configurations when examining these options.

Meta SAN

A *Meta SAN* is a collection of fabrics that are connected together through some routing mechanism. The routing mechanism can either be a single router or multiple routers, which are connected either by the FC or IP protocol. Examples of Meta SANs are: an edge switch connected through IFL to a EX_Port on a Brocade switch; fabrics connected by Cisco's IVR-NAT; a single Brocade M Series 2640, or two Brocade M Series 2640s connected by an MFCP link; fabrics or Meta SANs connected by an iFCP link; two Brocade fabrics connected by VEx_Ports; or a transit Cisco VSAN which spans an IP link.

iSCSI SANs

iSCSI Storage Area Networks exclusively use the iSCSI protocol over TCP/IP for the transport of command, data, and status information.

IP is a network layer protocol that provides datagram routing services for transport layer protocols such as Transmission Control Protocol (refer to "Transmission Control Protocol (TCP)" on page 333) and UDP (User Datagram Protocol). Datagrams are the packets that carry user and application data end-to-end across router-connected networks. A TCP/IP network can actually be viewed as a communications medium designed to transport IP packets.

For more information on IP SANs, refer to Chapter 4, "IP SAN Concepts."

Multiprotocol SANs

Multiprotocol Storage Area Networks have some elements that use Fibre Channel or iSCSI for the transport of command, data, and status information. This class of SAN also includes ISANs formed by spanning FCIP and iFCP links as well as FC SANs that have iSCSI initiators accessing FC targets through a bridge.

iFCP

iFCP, defined in RFC 4172, specifies an architecture and a gateway-to-gateway protocol for the implementation of Fibre

Channel fabric functionality over an IP network. This functionality is provided through TCP protocols for Fibre Channel frame transport and the distributed fabric services specified by the Fibre Channel standards. The architecture enables internetworking of Fibre Channel devices through gateway-accessed regions with the fault isolation properties of autonomous systems and the scalability of the IP network.

iFCP supports FCP, the ANSI SCSI serialization standard to transmit SCSI commands, data, and status information between a SCSI initiator and SCSI target on a serial link. iFCP replaces the transport layer with an IP network (Ethernet), but retains the upper layer information such as FCP.

In an EMC environment, iFCP would allow Fibre Channel switches to use IP as the interswitch *fabric* protocol. This would allow customers to utilize existing IP infrastructure (cabling and switches, for example) to support storage-to-storage communications.

I/O convergence with FCoE

I/O consolidation has been long sought by the IT industry to unify the multiple transport protocols in the data center. This section provides a basic introduction to Fibre Channel over Ethernet (FCoE), which is a method to achieve I/O consolidation that was originally defined in the FC-BB-5 T11 work group.

I/O consolidation, simply defined, is the ability to carry different types of traffic, having different traffic characteristics and handling requirements, over the same physical media. I/O consolidation's most difficult challenge is to satisfy the requirements of different traffic classes within a single network. Since Fibre Channel is the dominant storage protocol in the data center, any viable I/O consolidation solution for storage must allow for the FC model to be seamlessly integrated. FCoE meets this requirement in part by encapsulating each Fibre Channel frame inside an Ethernet frame.

The goal of FCoE is to provide I/O consolidation over Ethernet, allowing Fibre Channel and Ethernet networks to share a single, integrated infrastructure, thereby reducing network complexities in the data center. An example is shown in Figure 3 on page 27.

FCoE consolidates both SANs and Ethernet traffic onto one Converged Network Adapter (CNA), eliminating the need for using separate Host Bus Adapters (HBAs) and Network Interface Cards (NICs).

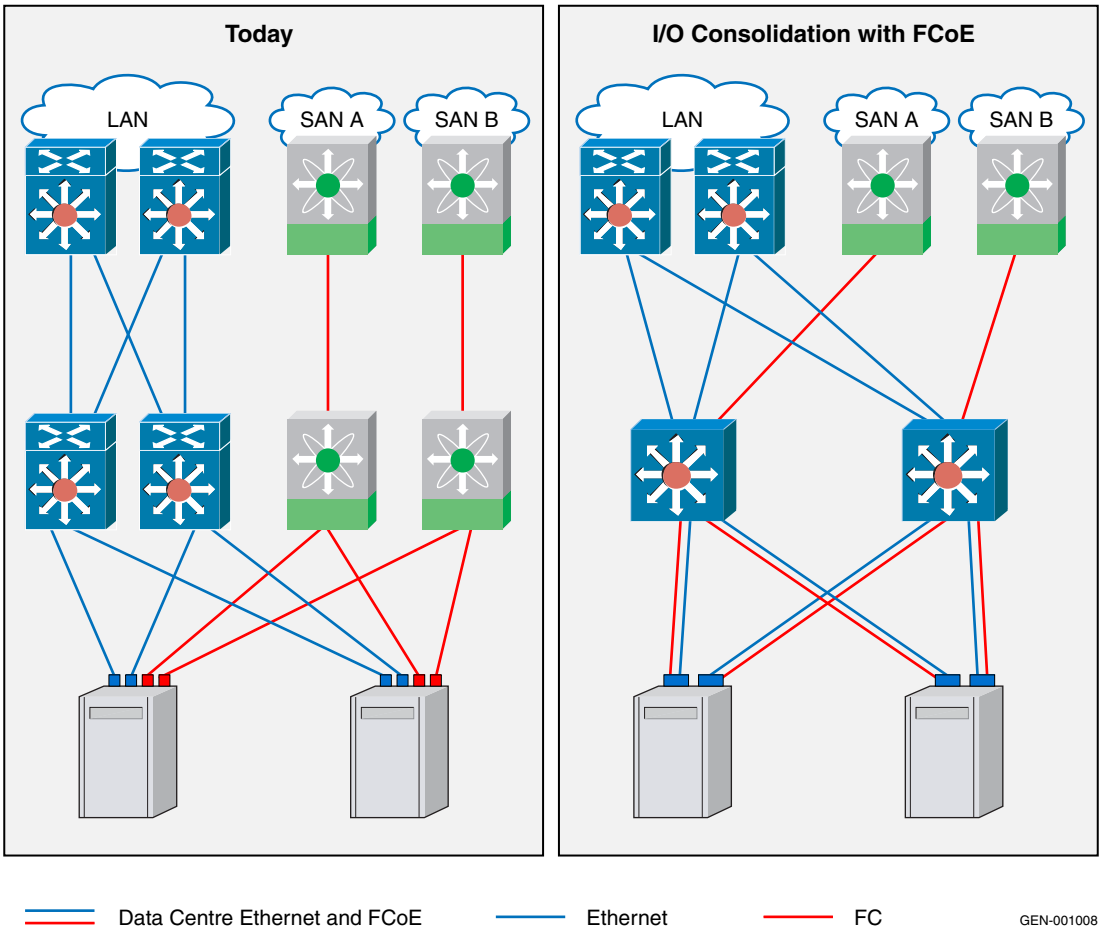


Figure 3 Typical topology versus FCoE example using Nexus 5020

- For more information on FCoE, refer to the *Fibre Channel over Ethernet (FCoE) Data Center Bridging (DCB) Concepts and Protocols TechBook* and the *Fibre Channel over Ethernet (FCoE) Data Center Bridging (DCB) Case Studies TechBook,* located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

Physical versus logical topologies

The Fibre Channel environment consists of a *physical* topology and a *logical* topology. The *physical* topology describes the physical interconnects among devices (servers, storage, and switch in the EMC®-specific environment). The *logical* topology describes the logical paths established between the operating system device names and their associated storage ports and volumes. Physical and logical topologies are further described in this section.

In Figure 4, the solid lines and their connections represent the physical topology and the dotted lines represent the logical topology.

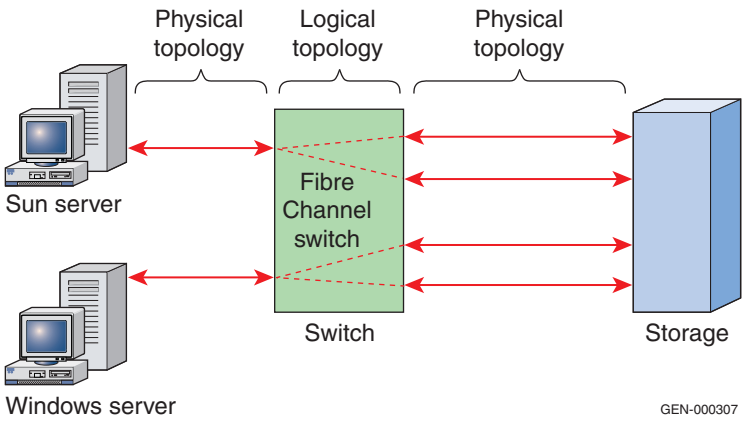


Figure 4 **Physical and logical topologies**

Physical topology

This section describes the details of the individual topology representations and shows how a logical topology is overlaid onto a physical topology. The physical topology can be described as actual hardware components in a fabric and the Fibre Channel cabling that interconnects them. A physical topology also includes the geographical locations of the switches and distances between them.

Some examples of the components and concepts used to describe the physical topology of a fabric are:

- Number of switches in the fabric
- Number of hops between any two switches
- Number of ports per switch

- Number of ISLs between switches
- Physical distance between any two switches

When describing a particular physical topology, it can be discussed in terms of its number of *tiers*. The number of tiers in the fabric is based on the number of switches that are traversed between the farthest two points in the fabric. It should be noted that this number is based on the infrastructure constructed by the fabric topology and does not concern itself with how the storage and server are connected across the switches.

Increasing the number of tiers in a fabric also increases the distance that a fabric management message must travel to reach every switch in the fabric. Increasing that distance can affect the time it takes to propagate and complete a fabric reconfiguration event (for example, adding a new switch), or zone set propagation event. Figure 5 displays one-tier, two-tier, and three-tier physical fabrics.

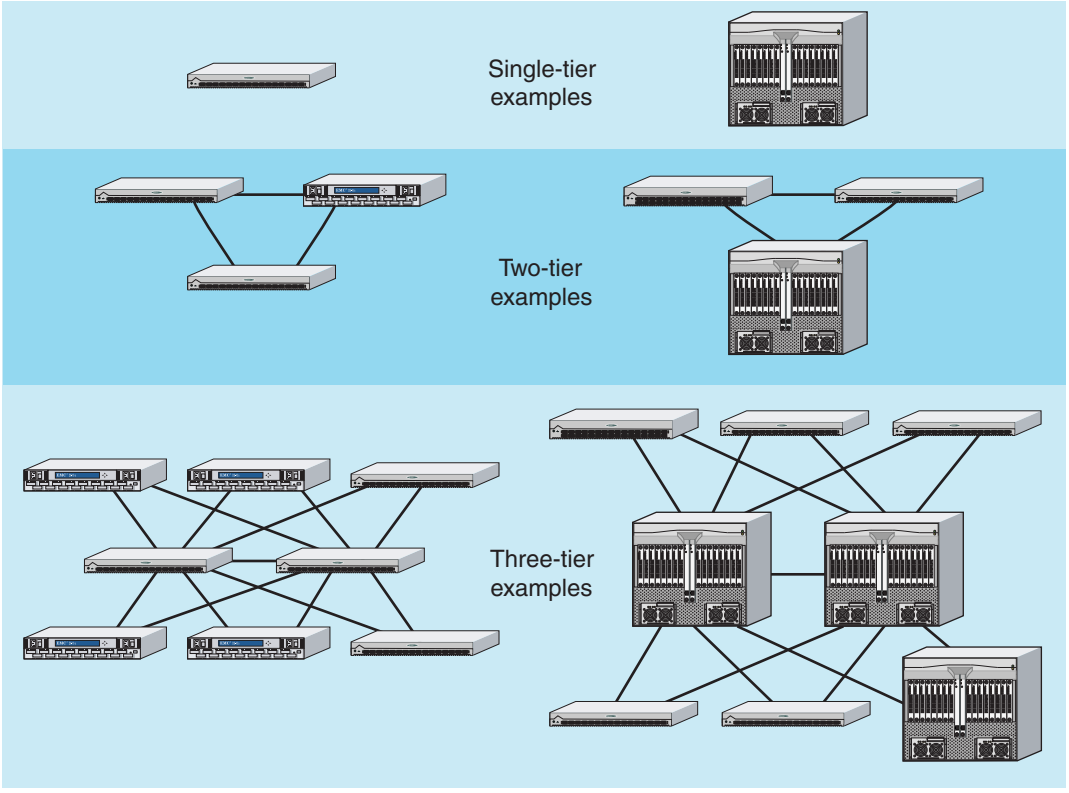


Figure 5 Tiered fabrics

As Figure 5 shows, a single-tier physical topology has a single switch. A two-tier topology has up to two switches between any two endpoints in the fabric. A three-tier topology has up to three switches between any two endpoints in the fabric. Currently, EMC recommends that the size of the fabric not exceed three hops, which equates to a four-tier physical fabric topology.

Final identification of your physical topology and later expansion of that topology relies on your ability to not only understand the individual impacts of the issues mentioned in this section, but also your selection of data protection schemes, logical topology, and management paradigm.

Core/edge fabric

Another common fabric design is the *core/edge* fabric. A core/edge design is based on the assumption that there will always be more host ports than storage ports and that providing equal and deterministic access to the storage from anywhere in the fabric is a marketable benefit to fabric management and storage administrators.

A core/edge design is built by consolidating storage access into a centrally accessible pool at the logical center of the fabric. From this *core* you can attach as many edge switches as necessary to service the hosts that require access to this storage. Each edge switch will be connected to each core switch, maintaining the fabric's accessibility as well its robustness. The number of core-to-edge ISLs can be based on the individual bandwidth requirements from each edge switch to the respective core switches or on overall fabric balance and flexibility. As described under "Methodology 1: Balanced fabrics" on page 39, balancing a fabric allows you easier management of component placement as the fabric requirements increase.

Since hosts do not have to communicate with other hosts over the Fibre Channel storage infrastructure, you can eliminate the need to connect edge switches to other edge switches over ISLs. Eliminating ISLs can greatly increase the number of host and storage ports that can be attached to the fabric. A simple core/edge fabric is designed to provide all hosts with single-hop access to all storage in the fabric.

While you strive to keep all hosts out at the edges, the attachment of excessively active hosts at the core to reduce some of the ISL bandwidth requirements is not allowed for. Core-to-core ISLs are added to facilitate RDF traffic, fabric management traffic, and backup traffic.

The core of the fabric can be extended to accept new storage access. EMC recommends that the storage core be built out as a full mesh to

perpetuate multiple paths to the storage, multiple paths for fabric management, and shortest-path access to all switches in the fabric.

No matter what fabric topology you are using in your design, you can still use the methodologies defined by mirroring fabrics or balancing fabrics, or use the steps outlined in this section on how to lay out hosts, storage, and ISLs to further enhance the accessibility, availability and manageability of your fabric.

Figure 6 demonstrates the topologies of single-core design.

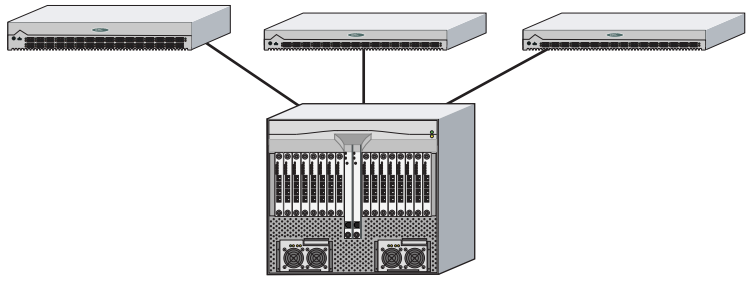


Figure 6 Single-core fabric

EMC recommends that single-core fabrics should always employ the fabric mirroring methodology to enhance the protection and robustness of the entire environment. While Figure 6 shows a director-class switch at the core and departmental switches at the edges, any switch type may be used, based on your connectivity needs.

Figure 7 demonstrates the topologies of a dual-core design.

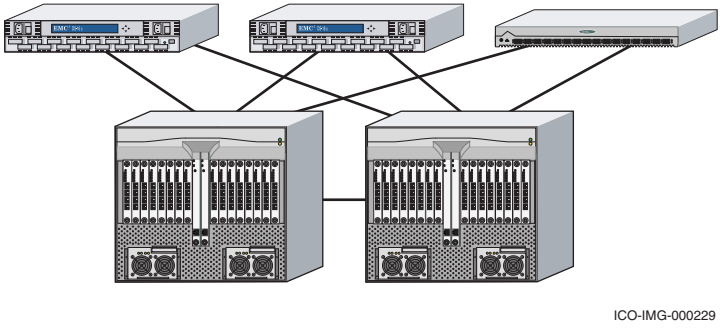


Figure 7 Dual-core fabric

Dual-core fabrics provide the ability to distribute mirrored storage ports across the core switches without going to a mirrored fabric protection scheme. (However, mirrored schemes provide even greater protection and isolation against events in the fabric.) Dual cores can also be expanded to contain more storage core switches, but for each core switch that is added, each edge switch will need ISLs to that core switch to maintain the topology.

Benefits

While the core/edge does not normally provide any zero-hop storage access, it does provide one-hop storage access to all storage in the system. Because traffic also travels in a deterministic pattern (from the edge to the core), a core/edge provides easier calculation of ISL loading and traffic patterns.

Since each tier's switch is used for either storage or hosts, you can easily identify which resources are approaching their capacity and arrive at an easier set of rules for scaling and apportioning. For example, if you are running out of available ports on our host switch, you know that if you are planning to expand your host numbers you must order the appropriate number of switches to handle the new hosts with their HBAs, as well as the ISLs from the edge to the cores to support the number of new switches.

A well-defined, easily reproducible building block approach makes rolling out new fabrics easier. Core/edge fabrics can be scaled to larger environments by linking core switches, adding more core switches, or adding more edge switches. This method can be used to extend the existing simple core/edge model or to expand the fabric into a compound or complex core/edge model.

Limitations

As the number of cores increases, it may be prohibitive to continue to maintain ISLs from each core to each edge switch. When this happens, EMC recommends shifting your fabric design to a compound or complex core/edge design, discussed next.

Logical topology

In contrast to a fabric's physical topology, a logical topology is concerned with where the Fibre Channel elements are attached around the fabric and the relationships (zoning) that define how these elements will be used together.

When describing a fabric's logical topology, it can also be discussed in terms of its number of logical tiers. The number of logical tiers in the topology is based on the number of switches traversed between a server and the storage zoned to it.

Since servers and storage can be located anywhere in the fabric, there could be several different logical tier relationships in the same fabric. Figure 8 demonstrates an example of multiple logical tier relationships in the same fabric.

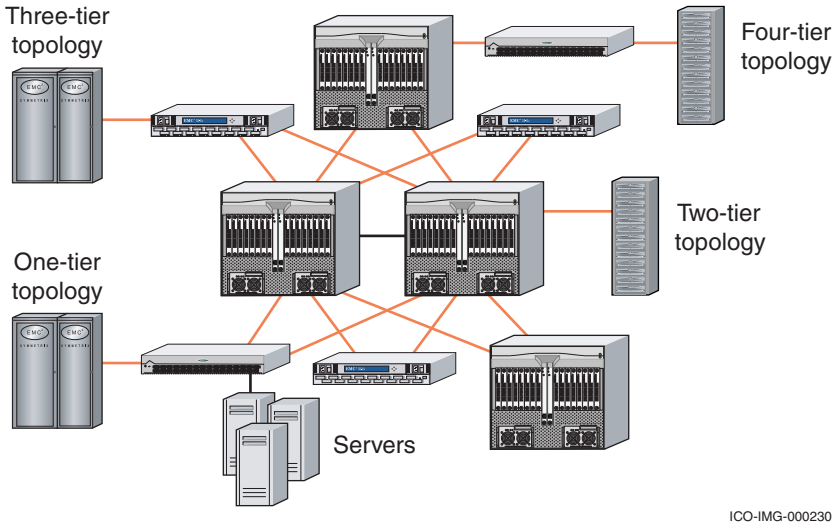


Figure 8 **Multiple tiers in the same fabric**

Note: As Figure 8 shows, the physical fabric size might not directly correlate to the logical fabric size.

Increasing the number of logical tiers in a fabric also increases the distance the data must travel from storage to the server. Each switch that the data has to traverse, and the length of the links between them, adds to the latency involved in sending or retrieving the data. Increasing the size of the logical fabric also increases the probability that bandwidth will be aggregated across the tiers. Excessive aggregation of traffic across the physical tiers can lead to fabric congestion and increased data retrieval latencies.

EMC recommends that you limit the path between the storage and the servers that are zoned to them to three hops. As Figure 8 shows, you can construct a four-tier logical fabric with three hops.

Logical topologies in the EMC/Fibre Channel switch environment can generally be described in terms of *fan-in* (*into* the EMC storage array) and *fan-out* (*out of* the EMC storage array). The example in Figure 4 shows a fan-in rate of 1:2 for each server. Refer to "Fan-in and fan-out considerations" on page 111 for more information.

Note: You can find some recommended fan-in and fan-out ratios on E-Lab Navigator.

The logical connectivity topologies, identified in Table 1, have been developed to solve three distinct customer problems. Each of these will be discussed in this section.

Table 1 Fibre Channel topology solutions

Problem	Solution
Proximity extension	Distance topology — Uses shortwave-to-longwave conversion to extend server-to-storage distance beyond shortwave's 500-meter limitation. Also includes DWDM solutions. (Refer to "Distance topology" on page 34.)
Capacity expansion	Capacity topology — Expands the storage capacity supported per host port by allowing a host port to connect to two or more Symmetrix® nodes. (fan-in). (Refer to "Capacity topology" on page 35.)
Storage consolidation	Consolidation topology — Expands the number of servers supported per Symmetrix port (fan-out). (Refer to "Consolidation topology" on page 36.)
More than one of the above problems	Combined topologies — For example, if you combine all three topologies you get high-capacity, highly available, multi-server, geographically dispersed clusters with a minimum of host I/O slots. (Refer to "Combined topologies" on page 37.)

Distance topology

Most Fibre Channel equipment is designed for shortwave lasers with multimode optical fibers, which is effective for up to 500 meters. Greater distances can be achieved using longwave lasers with single-mode fiber.

Symmetrix Fibre Channel directors support distances up to 500 meters with 50/125 fiber cable. By using the appropriate longwave adapter and 9/125 single mode cable, the distance can be extended to 10 km from one Symmetrix port to a second longwave Symmetrix port or switch port. (Refer to Figure 9 on page 35).

Note: Although the Symmetrix uses optics that are capable of 10 km, links over 7 km will experience a slight droop in throughput when running at 2 GB/s due to the number of BB_Credit each FA provides. See "Determining customer requirements" on page 91 for more information.

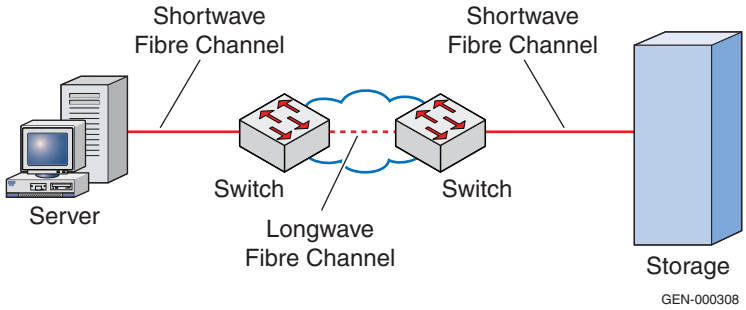


Figure 9 FC-Switched fabric distance topology example

Greater distances require two fabric switches connected with longwave lasers over 9/125 single-mode fiber cable. The switches interconnect through a port called an *expansion* port (E_Port) over a connection that is referred to as an interswitch link (ISL).

Another option for distance extension requires a Dense Wavelength Division Multiplexing (DWDM) system.

Another distance extension technique is to use either iFCP or a similar IP-based solution. The advantages for using an IP based solution over distance are error detection and the ability for the IP gateway to request the re-transmission of lost packets.

Capacity topology

This topology expands capacity in the Symmetrix environment, allowing a single host bus adapter in a file server to access a large capacity that might be stored on multiple Symmetrix devices.

The logical concept of the capacity topology in a switched fabric is described by the *fan-in* rate (*in* as in *into* the storage system).

The capacity topology uses a logical topology of fan-in, as shown in Figure 10 on page 36, which illustrates a fan-in rate of four. Each storage port in the figure connects to a shared server port.

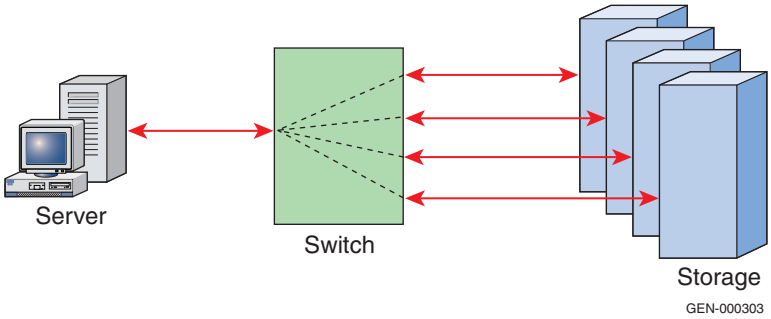


Figure 10 FC-Switched fabric capacity topology example

In Figure 10, assume that several host I/O slots are needed for tape, clustering, and networking connections, leaving only one slot in the host available for attachment to four storage ports. By using a Fibre Channel, host bus adapter and a Fibre Channel switch, the host can connect to a large pool of storage, minimizing the server I/O slot requirement.

Consolidation topology

Some environments contain low-capacity servers that must be connected to a high-capacity storage, expanding the required number of server connections. The consolidation topology provides a solution for this situation.

Both clustered and non-clustered applications are possible. Through host-based file-locking facilities, clustered hosts share information assets. Non-clustered application environments can share physical storage assets, capacity, bandwidth, and connectivity.

Channel failover and channel load-balancing software products (like PowerPath) can play a central role in providing manageability, reliability, and performance benefits.

Consolidation topology in the fabric environment

The logical concept of a consolidation topology in a switched fabric is described by the *fan-out* rate (*out* as in *out of* the storage system).

The consolidation topology uses a logical topology of fan-out. Figure 11 on page 37 illustrates a fan-out rate of four, where four server connections share a single storage port.

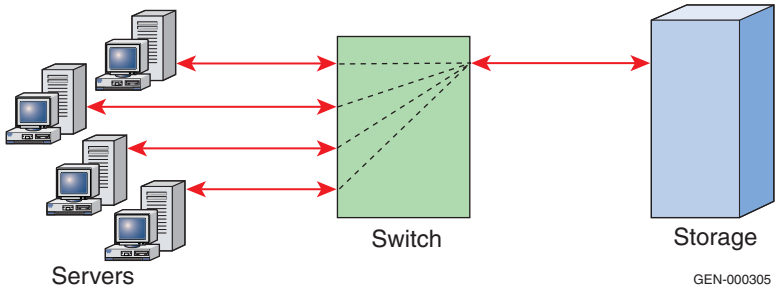


Figure 11 **FC-Switched fabric consolidation topology example**

Combined topologies

Topologies can be combined for maximum efficiency, achieving large storage capacity for many servers while minimizing the necessary number of host bus adapters and storage ports. Figure 12 shows how the three basic topologies can be combined to take advantage of the benefits of each.

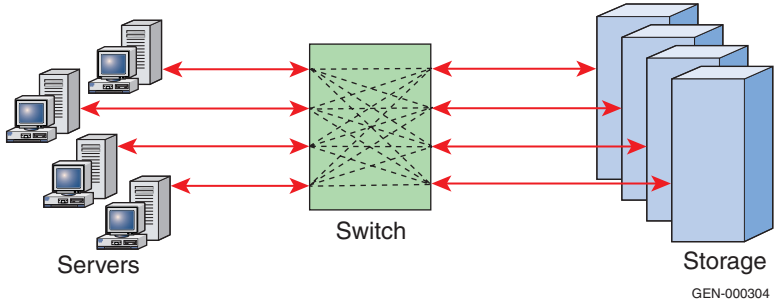


Figure 12 **FC-Switched fabric combined topologies example**

This type of configuration can take advantage of PowerPath and the redundancy of dual switches to provide alternative routes if necessary.

Combing physical and logical topologies

In the previous sections, the various physical and logical topology concepts were introduced. This section combines different pieces from of each of these topology concepts to highlight different design methodologies. As will be discussed, each methodology has its strengths and weaknesses.

The section starts with an introduction to the basic types of traffic that are present in a fabric and then highlights four different methodologies for managing these traffic patterns.

An example of a combined physical and logical topology is shown in Figure 13, which displays a three-tier physical fabric that is also configured as a three-tier logical fabric. There are two hops between the server and the storage that is zoned to it. This fabric also has two separate same-cost, same-distance (hop) routes, indicated by the heavy blue lines. The thin black lines indicate ISLs that are in the fabric but are not currently being used for traffic. These unused ISLs are also referred to as *cold* secondary ISLs.

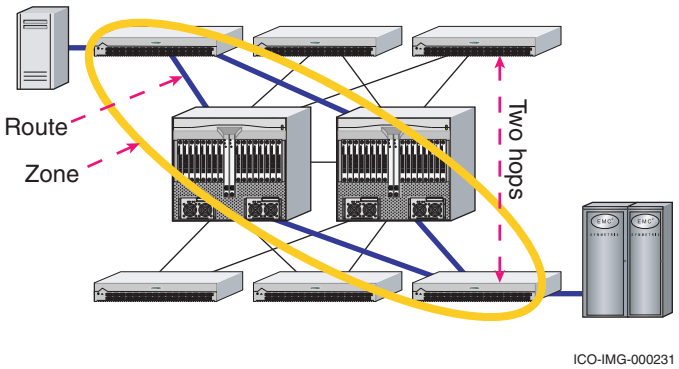


Figure 13 Three-tier physical and logical fabric

Traffic types

Note the following types of traffic regions that can be constructed to handle specific traffic and tasks.

Subscriber traffic

Subscriber traffic refers to that traffic associated with data being transferred between the application hosts and their storage.

Management traffic

Management traffic refers to that traffic associated with switch-to-switch traffic necessary to manage the fabric topology,

user-generated traffic, or traffic from the management workstations to the storage arrays or fabric controllers for the purpose of configuring the environment. This includes:

- Zoning changes
- Volume access/masking changes
- Fabric build messages

Business continuance traffic

Business continuance traffic refers to the data traffic generated through the use of a data replication application (SRDF or MirrorView) or the action of backing up, restoring, recovering, or retrieving files from a removable (tape) or near-line (digital vault/virtual tape) storage system.

Methodology 1: Balanced fabrics

Balanced fabrics, or balanced environments, are constructed by distributing the fabric resources (switches and ISLs), storage, and servers evenly across the environment so that the bandwidth and usage are also evenly distributed across each switch and ISL. No matter which specific fabric topology is chosen, you can create a balanced environment.

The first step in creating a balanced environment would be to create the fabric topology by laying out the switches in the desired pattern and attaching the ISLs evenly across the tiers of the fabrics. The aim is to create an infrastructure that allows equal bandwidth utilization for all points in the fabric. This design allows for better flexibility and more consistent performance as you increase the amount of shared storage access from your distributed hosts.

A further level of balancing can be accomplished by strategically using similar switch types (Connectrix® M Series or B Series) and/or similar technologies (2 Gb/s or 1 Gb/s) in the fabric. Using similar technologies across the fabric can standardize and simplify later troubleshooting efforts. If the fabric is constructed of mixed switch types, balance can still be achieved by strategic placement of the switch technologies across the fabric for best utilization of the unique features and balanced bandwidth utilization.

The number of ISLs used should be based on both the desired level of redundancy and the estimated level of bandwidth. To create the balanced environment, the bandwidth requirement estimates should be evenly distributed across the entire fabric.

Lastly, the storage, and then servers, would be attached according to their access requirements and zoning relationships so that the bandwidth requirements across each link would be managed evenly across all links.

In Figure 14, each line represents two ISLs. As shown, each switch has the same number of servers and, for the sake of the example, each server has the same amount of bandwidth requirements to each storage port. Storage ports would also be connected evenly across the storage level switches. Zoning and volume accessing would then be configured so that each link would carry an even portion of the total load. Fabric traffic would be constantly monitored and migrations of servers and storage could be made to perpetuate the level of bandwidth balance desired in the fabric.

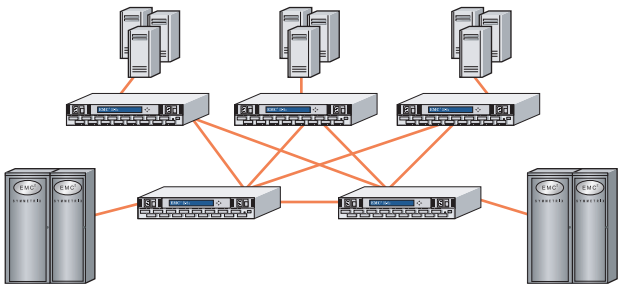


Figure 14 **Balanced fabric**

By creating the initial balance and maintaining the balance over time, you can produce an environment with predictable and deterministic traffic and fabric behaviors. Ensuring predictable fabric behavior facilitates identification of issues in the fabric components and may simplify troubleshooting. Because the data load is balanced across the ISL and switches, a more efficient use of fabric resources is achieved. Underutilized ISLs can be removed and free ports can be used for attaching more servers and storage.

Methodology 2: Mirrored fabrics

A mirrored fabric is an environment created by two or more distinct fabrics that, while they may share a common out-of-band management connection, are not directly connected to each other by fiber.

When mirroring an environment, start by configuring each physical device in the storage array so that it is presented across multiple

storage ports. These storage ports would then be distributed evenly across both fabrics. Each host accessing this storage would also have multiple ports distributed across both fabrics.

Volume access and device pathing to these storage arrays through the servers would be handled by load-balancing and path-failover software, such as PowerPath. PowerPath manages the representation of the multiple instances of the same devices to the host and manages the load balancing and path failover necessary to maintain the high availability of the environment.

Figure 15 shows how a server and storage array would be connected to both fabrics in a mirrored environment.

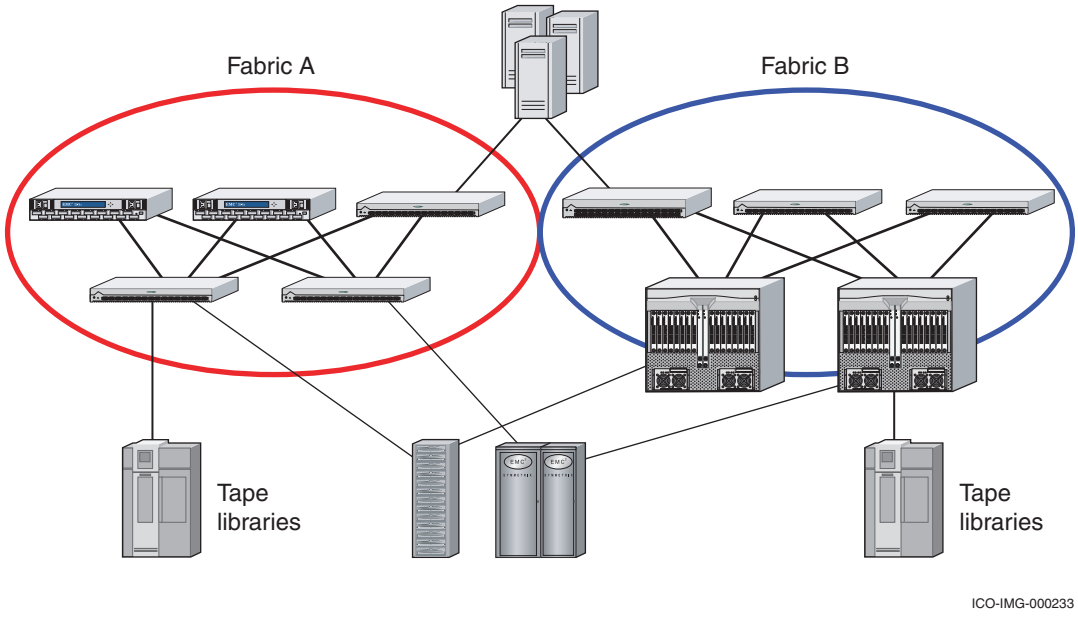


Figure 15 Mirrored fabrics

The decision to mirror a fabric can also be made independent of the individual topology used to create each mirror. Also, while EMC recommends that each side of the mirror be identical for easier management, monitoring, and performance characterizations, you can change the topology or switch vendor on one side of the mirror and phase in new technologies without any impact on the other side of the mirrored fabric.

Having different versions of firmware across a balanced fabric configuration does not pose any HBA, switch, or storage interoperability concerns, and is fully supported.

Balanced (redundant) fabric configurations can be created from any supported combination of switches. As long as the host and storage ports are supported with the switch firmware revision they will be connected to, there is no restriction on connecting host and storage ports to fabrics consisting of different vendor, model, or firmware revisions.

It is important to note that in homogeneous switch vendor environments, all switch firmware versions inside each fabric should be equivalent *except* during the firmware upgrade process.

For devices that have single attachment (such as tape or JBOD), EMC recommends that for the highest level of availability and distribution of workload, these devices be distributed evenly across both fabrics. Since each host in the environment has access to both sides of the fabric, connections to all devices are maintained. Backup environments can be configured so that device pools contain tape or disk devices from both sides of the mirrored environments.

To simplify maintenance and management, all additions to the environment should be made in pairs. For example, if you add storage access to Fabric A, ensure there is an identical addition is made to Fabric B.

Benefits of mirrored fabrics

Many individual benefits can be directly associated with creating a mirrored fabric environment. The following are some major benefits, but this should not be thought of as an exhaustive list.

- Increased availability through insulation and isolation — One of the most important benefits to implementing mirrored fabrics is the isolation your mission-critical applications will receive from the increased buffering from fabric events. Issues that arise from component failures will be totally isolated to the single fabric experiencing the event. Mirrored fabrics can also be separated by distance so that a local room or floor outage or event will not affect the entire environment.
- Simplification to increase availability — By separating the environment into equal parts, you are able to manage much smaller units more confidently and effectively. Problem isolation becomes easier as there are fewer components in the environment under investigation at one time. Also, problems that occur on one

side of the mirror are isolated from the other side of the mirror. Since each fabric is effectively a separate environment, you can maintain separate, smaller zone sets specific to each fabric.

- Increased effective domain counts in the environment — In general, EMC recommends that you limit your individual fabric designs to 31 domains or less switch topologies. By separating the required environment into two equal pieces, you increase the effective accessibility by having twice the number of switches and switch ports to work with.
- Increased N_Port availability — Since the number of switches is evenly distributed across two mirrored fabrics, the effective ISLs that would have joined the switches from both fabrics together can be freed up to attach more storage and servers. Reducing the infrastructure requirements associated with additional ISLs reduces the cost associated with the operation and management of the fabric.
- Flexibility — Since each fabric is independent of the other, you can perform maintenance and upgrade procedures on one side of the mirrored environment without affecting the other. The risk associated with updating firmware, recabling, implementing zoning, security, or other management changes, as well as physical movement of the devices, is greatly reduced because of the insulation benefits provided by mirroring fabrics.

  You may also use the isolation of the fabrics to phase in or leverage new hardware from different vendors, or hardware that uses more advanced technologies, with reduced risk to the entire environment. You may also choose to use the isolation to validate topology or management changes without affecting the entire environment. After changes are made to one side of the mirrored fabric, you can use the original design as a benchmark to contrast the performance of the changes.
- Centralized or distributed management — Management of both fabrics in the mirrored environment can be performed from either a centralized location or separate locations to further limit the dependencies. Management applications, such as EMC Ionix™ ControlCenter®, can be used across single or multiple Connectrix service processors across multiple Ethernet segments.

Methodology 3: Logical fabric segregation

Logical fabric segregation is a set of design concepts that you can use during the planning stages to optimize a single fabric environment. While you normally think of a fabric in its entirety, the fabric can be

logically segregated into separate traffic regions that are constructed to handle specific traffic and tasks. For example, a single fabric can be constructed so that subscriber traffic and business continuance traffic travel over the same fabric, yet over separate parts of the fabric.

Logically, segregating the fabric into different traffic centers limits the dependence of one storage task on another. Limiting the dependence makes management easier because the fabric resources can be designed, scaled, and maintained somewhat independently of one another. Troubleshooting is also simplified because the data traffic related to any one task may be relegated to only a specific portion of the fabric.

As Figure 16 on page 44 shows, the subscriber traffic occurs in the area indicated inside the larger oval. SRDF traffic, MirrorView traffic, and SAN backup traffic all occur within the area indicated by the smaller oval.

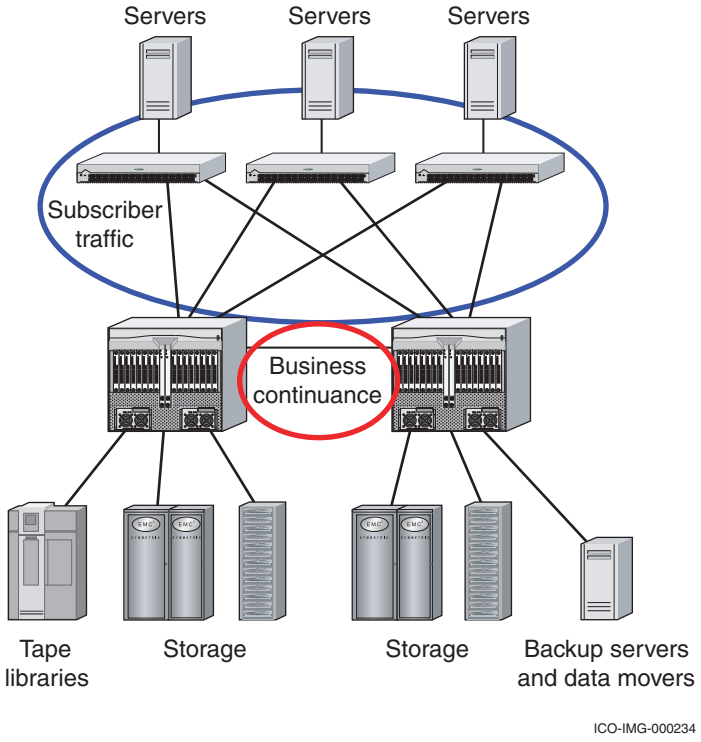


Figure 16 Logical fabric segregation

ISLs can be added or removed based on the specific bandwidth requirements of the given task.

Methodology 4: Business unit fabrics

Business unit fabrics are fabrics that are created to perform a particular function or process, or a set of business rules, bandwidth or availability requirements, support resources, or maintenance/duty cycles. Business unit fabrics can employ the benefits of both balancing and mirroring to further increase their robustness and availability.

Business unit fabrics are related to mirrored fabrics in the sense that they offer similar benefits in the isolation, limiting of dependencies of applications and processes on unrelated components, and the simplification of an environment. For example, one might implement a unique fabric for a line of business that uses highly available Fibre Channel directors.

That line of business might have special uptime requirements. The added expense to use the new directors might be justifiable for this particular line of business, but might not be for the entire environment. The port-count requirements for this single line of business will be smaller, and thus require fewer directors. The converse is also true; there may be lines of business that require less guaranteed storage access and their requirements may allow them to work on a lower bandwidth set of switches or different topology.

Separating fabrics based on business units also allows each business unit to manage their maintenance, duty cycles, and upgrade schedules based on their availability. It will be much easier for the storage manager to coordinate the changes with the groups separately, rather than as a collective.

Business continuance traffic (backup, recover and restore, and remote data replication facilities such as SRDF and MirrorView) has been a prime candidate for business unit fabrics because of the differences in duty and maintenance cycles and the independence from application data (subscriber traffic). Customers have also chosen to build smaller test fabrics that allow them to investigate and validate components and applications before they are implemented into their production environment.

If you are considering the use of business unit fabrics in your environment, consider the following questions:

- Are all projects created equally based on their requirements for bandwidth, availability, and accessibility?

- Do the individual duty cycles and maintenance cycles of my individual business units differ to the point that they hinder one's ability to do business as efficiently as possible?
- Are there interdependencies between how the storage or servers are shared between business units that cannot be decoupled?

2

Fabric Design Considerations

This chapter provides information to help you understand how to design and implement a Fibre Channel storage area network (SAN).

Fabric design considerations and recommendations

Note: Until this point, the material presented has been at the conceptual level and has been sparing in the actual design and setup detail. This section takes the first steps toward understanding how to design a solution. Throughout the next several sections, the design concepts will also have pointers to actual case studies containing cookbook setup steps.

This section describes the considerations and methodologies associated with the design and implementation of a Fibre Channel storage area network (SAN). The items detailed here represent a set of best practices that have been collected though SAN engagements with EMC customers and verified through testing in the EMC E-Lab.

While there is no substitute for tracking actual data traffic, application and storage utilization, and performance characterizations, there is still a need for determining a set of rules to approach the initial designing of a fabric. It is the objective of this section to outline some of the considerations involved in the designing of reliable fabrics and to provide some recommendations to managing a fabric's expansion over time.

It is imperative that after the initial fabric has been implemented, it be continually monitored to evaluate the success of the design and actual bandwidth utilization and to identify future needs for expansion.

To help design a fabric, the following topics are discussed in this section:

Fabric design considerations

Designing a fabric involves many variables that require consideration. With each variable consideration comes a separate design decision that must be made. Each design decision will help you create a fabric design that is appropriate for your business information model. This section discusses the following major considerations to aid you in evaluating how they are prioritized in your business:

Note: Each of the examples presented in "Common fabric topologies" have been assigned a subjective rating. The subjective rating grades each topology on how well it meets each of the major considerations previously listed.

Accessibility

Accessibility refers to the ability of your hosts to access the storage that is required to service their applications. Accessibility can be measured on your ability to physically connect and communicate with the individual storage arrays, as well as your ability to provide enough bandwidth resources to meet your full-access performance requirements. A storage array that is physically accessible, but cannot be accessed within accepted performance limits because of oversaturated paths to the device, may be just as useless as an array that cannot be reached physically.

Accessibility's link to available bandwidth leads us to consider the differences in building a statistical bandwidth infrastructure and a guaranteed bandwidth infrastructure. Guaranteed bandwidth infrastructures provide enough bandwidth resources for the full potential of the devices on the fabric to be used simultaneously. Statistical bandwidth fabrics are developed to handle only a fraction of the potential bandwidth.

An example of a statistical bandwidth network is the telephone system. The telephone system is not constructed with enough

bandwidth resources to allow every subscriber to communicate simultaneously. You may have heard "all lines are currently busy; please try your call again later." This message indicates that the number of subscribers has saturated the bandwidth currently available, so no new connections are possible until resources are freed. Similar issues can arise in the design and implementation of a fabric.

You should also consider the internal design of the switching devices used in your fabric when considering accessibility. While switches may be designed for high levels of connectivity and allow many physical attachments, their internal designs may cause internal bandwidth congestion. Even though you may develop a fabric topology that eliminates external congestion, your fabric may still be limited by the individual switches if they are designed with internal bandwidth congestion points.

Full *any-to-any* accessibility comes at a price. Increasing the accessibility of the fabric also increases the size of that fabric and the complexity of the topology. As more switches are needed to connect the new hosts and storage, more ports will be required for switch-to-switch communication. (Refer to "Determining customer requirements" on page 91 for scalability information.) You must be careful to ensure that our switch-to-switch bandwidth designs do not cause external congestion points in the fabric.

As the complexity of the fabric increases, more emphasis and reliance must be placed with the management application and adherence to well-defined management policies. Complexity can also impact how an environment is secured.

Availability

Availability is a measurement of the amount of time that your data can be accessed, compared to the amount of time the data is not accessible because of issues in the environment.

Lack of availability might be a result of failures in the environment that cause a total loss of paths to the device, or it might be an event that caused so much bandwidth congestion that the access performance renders the device virtually unavailable.

Availability is impacted not only by your choice of components used to build the fabric, but also by your ability to build redundancy into the environment.

The correct amount of redundancy will allow processes to gracefully failover to secondary paths and continue to operate effectively. Too

little redundancy built into the fabric can cause bandwidth congestion, performance degradation, or (in some cases) a loss of availability.

When considering building redundancy into the environment, you must always weigh the *opportunity cost* of how much redundancy you need/want to build into the environment. Opportunity cost is the cost associated with the possible impact of not using the resources for other activities. For example, each extra port that is used for redundant ISLs cannot be used to attach more servers and storage to the environment.

Another concept that adds to the availability of an environment is *sparing*, which is the process of dedicating resources to remain unused until they are needed to take the place of a failed resource. You might also develop a sparing plan that indicates that even though a resource is currently in use, it may be swapped for a failed resource that has a higher priority.

The following must be considered in your redundancy and sparing plan:

- How much bandwidth do I need to preserve after a single event occurs?
- What other applications might be affected when the original storage resources move to a new path or down to a single path?
- Do I need to plan for scenarios that include successive failures?
- Do I want redundancy built into my connectivity components (as seen with director-class switching devices)?
- Do I want to build site redundancy and copy data to another site using the Symmetrix Remote Data Facility (SRDF) or VNX™ series or CLARiiON® MirrorView?
- Do I want to build redundancy at the host level with a load-balancing and paths failover application (like PowerPath)?
- How do I rank my business applications so that I can identify lower priority tasks, so these resources can be used as spares during a failure event? An example of this would be if task one had failed due to all of its fiber links being damaged and fiber links from task two were used to bring up the resources associated with task one. When the resources were back online, both tasks would be working at 50 percent efficiency.

Consolidation

Resource consolidation includes the concepts of both physical and logical consolidation. Physical consolidation involves the physical movement of resources to a centralized location. Now that these resources are located together, you may be able to more efficiently use facility resources, such as HVAC (heating, ventilation and air conditioning), power protection, personnel, and physical security. The trade-off that comes with physical consolidation is the loss of resilience against a site failure.

Logical consolidation is associated with bringing components under a unified management infrastructure and creating a shared resource pool, such as a SAN. Logical consolidation does not allow you to take full advantage of the site consolidation benefits, but it does maintain site failure resilience.

Flexibility

Flexibility is a measure of how rapidly you are able to deploy, shift, and redeploy new storage and host assets in a dynamic fashion without interrupting your currently running environment. An example of flexibility is your ability to simply connect new storage into the fabric and then zone it to any host in the fabric. You can do this without any interruption in the I/O to any of the other hosts in your environment. Flexibility can also be seen in the ability of the Fibre Channel directors to perform code loads, component replacement, and insertion while the system is running without any noticeable impact to the hosts.

Scalability

Scalability is a measure of how easily a fabric can be extended so that it can accept more storage, more hosts, or more connectivity (switches). More storage and servers not only cover the physical connections required to attach these components, but also the internal and external bandwidth required to handle the actual throughput of these devices during usage.

External bandwidth is associated with the number of ISLs a switch can support, as well as the individual bandwidth of each ISL. Some switches now support both 2 Gb/s and 1 Gb/s ISLs. Other switches support a feature called *trunking*, which allows sharing the bandwidth resources of multiple ISLs simultaneously.

Fabric topologies that aggregate traffic into unbalanced scenarios should be avoided. One such scenario is when an intermediate switch has two ISLs coming in from the server-tier switches and only one ISL leaving toward the storage-tier switches. This scenario can lead to performance degradation, sometimes referred to as an *externally congested* architecture.

Each vendor's switch design has an associated internal bandwidth model that has a specific set of resources available to transfer data inside the switch. These internal resources may or may not aggregate bandwidth between the switching chips. Those architectures that do aggregate traffic inside the switch design may also represent an *internal congestion* architecture. Internal or external congestion architectures limit the scalability of the environment.

Scalability is enhanced by a switching component's ability to allow the online insertion of port expansion cards or additional optics. Allowing *hot* insertion of devices promotes the purchase and usage of partially-populated chassis that can be upgraded as the need arises. Partially-populated chassis can also play a role in the creation of redundancy in the fabric.

As the fabric grows, so does the ability of the switches to accept large numbers of zones and zone members. A fabric topology also plays a role in how quickly zoning or fabric changes take to propagate to all other switches in the fabric. Better fabric designs mean that your fabrics can recover from fabric events and propagate changes to the environment faster and with more predictability.

Security

Security refers to the ability to protect your operations from external and internal malicious intrusions, as well as the ability to protect accidental or unintentional data access by unauthorized parties.

Security can range from the restriction of physical access to the servers, storage, and switches by placing them in a locked room, or the logical security associated with zoning, volume accessing/masking, S_ID lockdown, or Port Binding.

Increasing the level of security directly impacts the flexibility of the environment. As security is increased, changes become more complicated. Whenever a new security policy is desired, it should be documented and reviewed for its impact on the accessibility, flexibility, and supportability of the environment.

For more information on security, refer to the *Building Secure SANs TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

Supportability

Supportability is the measure of how easy it is to effectively identify and troubleshoot issues, as well as to identify and implement a viable repair solution in the environment. The ability to troubleshoot may be enhanced through good fabric designs, purposeful placement of

servers and storage on the fabric, and a switch's ability to identify and report issues on the switch itself or in the fabric.

Fabric topologies can be designed so that data traffic patterns are deterministic, traffic bandwidth requirements can be easily associated with individual components, and placement policies can be documented so that troublesome components can be identified quickly.

The supportability measurement takes into effect the usefulness of internal error reporting, logging, and any diagnostic utilities that are shipped with the component. A product that is easily supported is an asset to an organization because of its ability to be brought back online without the time delays associated with shipping replacements or from scheduling visits for onsite service representatives. Many switches have the ability to identify issues and, through policy management, initiate automatic failover and recovery procedures. Some switches, as well as the Symmetrix system, also have the ability to identify issues and initiate call-home procedures to alert support personnel of the issue.

Common fabric topologies

This section provides information and recommendations on the following types of common fabric topologies.

For simple Fibre Channel SAN topologies:

- "Single switch fabric" on page 55
- "Two switch fabric" on page 56

For complex Fibre Channel SAN topologies:

- "Full-mesh fabric" on page 57
- "Compound core/edge fabric" on page 59
- "Partial mesh fabric" on page 62
- "Connectivity tier fabric" on page 64

Refer to the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab, for more detailed implementation information for simple and complex topologies.

Simple Fibre Channel SAN topologies

A *simple* Fibre Channel SAN consists of less than four directors and switches connected by ISLs and has no more than two hops.

Single switch fabric

A single switch fabric is the simplest of the simple Fibre Channel SAN topologies and consists of only a single switch (Figure 17). The switch is also connected to a single management LAN through IP. For more information on single switch topology, refer to the "Single switch fabrics" section in the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

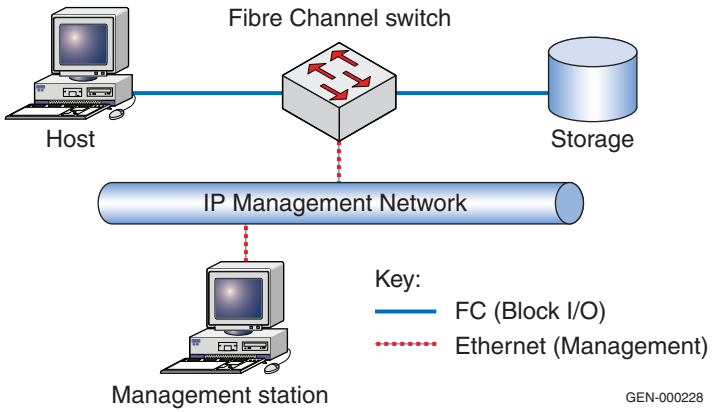


Figure 17 Single switch topology example

Table 2 shows the subjective rating for a single switch.

Table 2 Single switch subjective rating

	Most <----------> Least				
Attribute	5	4	3	2	1
Accessibility					
Availability					
Consolidation					
Flexibility					
Scalability					
Security					
Supportability					

Each of these attributes are discussed in more detail in the "Fabric design considerations" section beginning on page 49.

Two switch fabric

Figure 18 on page 56 shows an example of a two switch fabric, which is slightly more complicated than the single switch fabric. The switches are connected using two ISLs.

Both switches are connected to Management Network A. For more information on two switch topology, refer to the "Two switch fabrics" section in the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

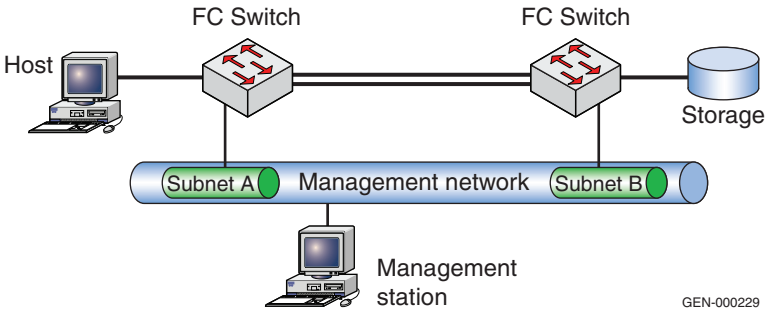


Figure 18 Two switch fabric example

Table 3 shows the subjective rating for a single switch.

Table 3 Two switch subjective rating

	Most <----------> Least				
Attribute	5	4	3	2	1
Accessibility					
Availability					
Consolidation					
Flexibility					
Scalability					
Security					
Supportability					

Each of these attributes are discussed in more detail in the "Fabric design considerations" section beginning on page 49.

Complex Fibre Channel SAN topologies

As the fabric size continues to grow, not all hosts will need one-hop access to all available storage. You may have a need to increase the storage and host connectivity while maintaining an efficient limit on the number of ISLs used in the fabric. You may also be searching for a way to add more storage accessibility to your hosts without major

changes to your fabric design. These factors are all considered in the design of both a compound and a complex core/edge fabric.

A complex Fibre Channel SAN consists of four or more directors and switches connected by ISLs and has any number of hops.

Full-mesh fabric

Table 4 shows the full mesh subjective rating.

Table 4 Full mesh subjective rating

	Most <----------> Least				
Attribute	5	4	3	2	1
Accessibility					
Availability					
Consolidation					
Flexibility					
Scalability					
Security					
Supportability					

Each of these attributes are discussed in more detail in the "Fabric design considerations" section beginning on page 49.

A *full mesh* fabric is any collection of Fibre Channel switches in which each switch is connected to every other switch in the fabric by one or more ISLs. For best host and storage accessibility, EMC recommends that a full mesh fabric contain no more than four switches. If you need to consider exceeding the four switch recommendation, you should first investigate the other switch topologies described in this section.

A mesh may contain departmental switches, directors, or both, depending on your connectivity needs.

EMC recommends that when designing and implementing a full mesh fabric you lay out the storage and servers in a single-tier logical topology design and plan your ISL requirements based on the assumption that 50% of the traffic on any one switch will remain local and the other 50% will originate from the remaining remote switches. This allows you a higher level of initial growth and resources sharing and gives you the following rule for optimizing your fabric performance:

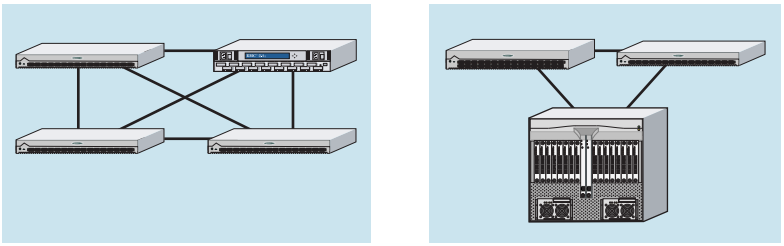
> If you realize that more than 50% of your traffic is originating from remote switches, it may be time to migrate those components exceeding the remote access requirements to other

switches so that they are either on the same switch or using a switch that is not fully utilizing its allocated switch-to-switch bandwidth.

If the initial remote storage access requirements are known, care should also be taken to distribute this load evenly across the remaining switches to equalize bandwidth utilization in the initial design. Understanding your initial storage-to-host fan-out and relationships can greatly increase your success in implementing a design that will be subject to minimal flux.

Identification of the highest bandwidth hosts and ensuring that they remain local to their storage needs will also ensure efficient use of the available switch backplane bandwidth before utilizing your ISL resources.

Figure 19 shows examples of full-mesh fabrics.



ICO-IMG-000235

Figure 19 Examples of a full mesh

Benefits

Full-mesh configurations give you, at most, one-hop access from any server to any storage device on the fabric. This means that when you are adding or migrating storage or server attachments you have the greatest possibility of placing the server attachment and matching storage attachments anywhere in the fabric and achieving the same response time. Meshes also ensure that you always have multiple local and remote paths to the data even after fabric events have occurred.

No matter which building block you choose for your fabric, you must be mindful in planning the traffic patterns and loading to ensure that you are not exceeding the technology's ability to handle the traffic.

For more details on designing a fabric that can handle your data traffic, review these topics:

- "Layout management" on page 67
- "ISLs in the fabric" on page 79
- "Trunking" on page 207

Limitations

Scaling a full-mesh solution becomes complicated and costly with the increase in the number of switches and required ISLs to guarantee traffic performance. Without rules for sharing access to storage on remote switches or an ability to migrate components once traffic increases, ISL bandwidth can become overutilized.

Compound core/edge fabric

Table 5 shows the shows the compound core/edge subjective rating.

Table 5 Compound core/edge subjective rating

	Most <----------> Least				
Attribute	5	4	3	2	1
Accessibility					
Availability					
Consolidation					
Flexibility					
Scalability					
Security					
Supportability					

Each of these attributes are discussed in more detail in the "Fabric design considerations" section beginning on page 49.

A compound core/edge fabric can be formed by merging two or more simple core/edge fabrics into a single fabric environment. Initially, hosts would be assigned volume access rights to storage that was available one hop away. Core switches that are one hop away from an edge switch in a compound or complex core/edge fabric are known as *primary core* switches. A core switch that is a primary core switch for one edge switch may also be a secondary core switch for an edge switch that is two physical hops away. Figure 20 on page 60 shows an example of a compound core/edge fabric.

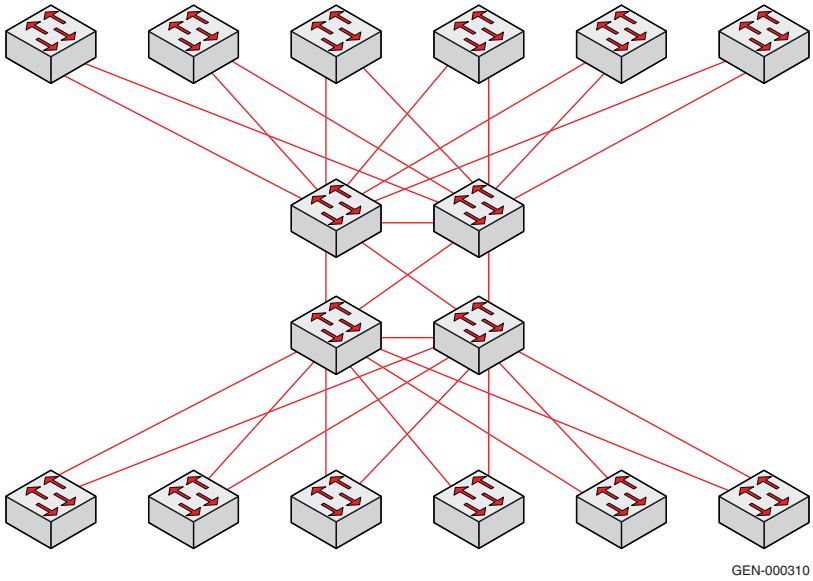


Figure 20 **Compound core/edge fabric example**

Once storage is exhausted for an edge switch's primary core switches, these hosts would then be assigned to storage that was two hops away on the secondary core switch. Access to the shared storage traffic on a secondary core switch would traverse the ISLs at the *back end* of the fabric. ISL planning for the compound core would be based on the both the subscriber traffic requirements at the *front end* of the fabric and the core-to-core bandwidth requirements at the back end of the fabric. Note that the front-end ISL requirements are independent of the back-end ISL requirements. This independence ensures a more efficient approach to ISL planning and utilization.

To facilitate the performance of fabric management traffic, the principal switch would be assigned (and remain assigned) to those switches at the core of the fabric.

Benefits

The compound core/edge model maintains a robust, highly efficient traffic model while reducing the required ISLs and thus increasing the available ports for both storage and host attachments. It also offers a simple method for the expansion of two or more simple core/edge fabrics into a single environment.

By connecting the core switches from simple core/edge fabrics into a full mesh, you can easily create a compound core topology.

The compound core/edge topology creates a robust back-end fabric that can extend the opportunities for sharing of both backup and RDF resources.

The complex core/edge fabric inherits the benefits from both the simple core/edge and the compound core/edge designs. The complex core/edge increases overall fabric availability by limiting the effects on the edge switches from multiple failures on the core switches.

Since the edge switches are more evenly distributed, a failure of any two core switches would result in fewer accessibility impacts to edge switches and attached hosts.

Further availability can be added by spreading hosts across edge switches that are not connected to the same set of core switches. Note that switch failures are rare and unexpected. Multiple simultaneous failures of Fibre Channel components are rarer still.

Limitations

Both the compound and the complex core/edge design models produce a physically larger, tiered fabric which could result in slightly longer fabric management propagation times over smaller, more compact designs. Neither compound or complex core/edge fabrics provide for single-hop access to all storage.

While the potential availability of the complex core fabric is increased over other designs, the designs are more complex and may add to management and troubleshooting time if care is not taken to document the environment. Management concerns would be eased by the implementation of a comprehensive fabric management infrastructure.

Partial mesh fabric

Table 6 shows the partial-mesh subjective rating.

Table 6 Partial mesh subjective rating

	Most <----------> Least				
Attribute	5	4	3	2	1
Accessibility					
Availability					
Consolidation					
Flexibility					
Scalability					
Security					
Supportability					

Each of these attributes are discussed in more detail in the "Fabric design considerations" section beginning on page 49.

A *partial mesh* fabric is different from a full mesh in that each switch does not have to be connected to all other switches. However, to be considered a partial mesh, the fabric must be of a configuration where splitting it results in each new sub-fabric being a full mesh. Figure 21 provides an example of partial mesh fabric.

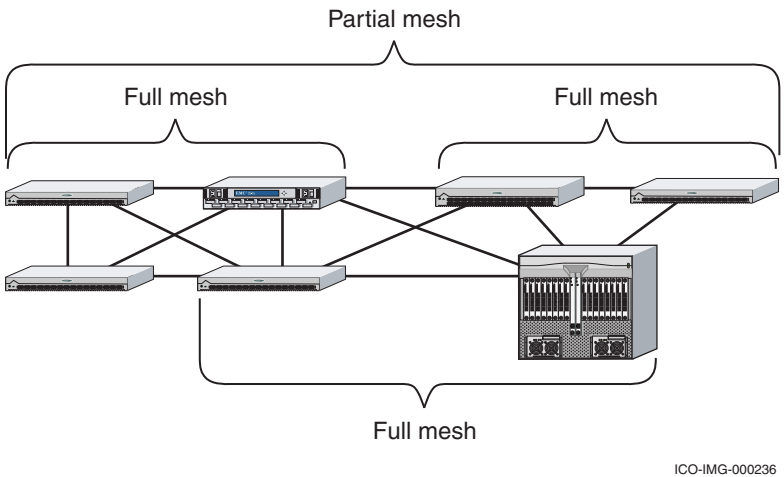


Figure 21 Partial mesh example

EMC recommends that you use the same rules for developing a partial mesh as for a full mesh. Storage and hosts would be placed in the fabric starting with a single logical tier approach, and additional storage access requirements would be placed one hop away.

Identifying the best selection from the pool of available storage should be evaluated based on your company's sharing, maintenance, and support procedures.

Servers with a majority of remote storage and storage that is heavily shared throughout the entire partial mesh should be moved toward the middle of the mesh. This should alleviate congestion and simplify pathing requirements evenly across the fabric.

EMC recommends limiting the size of the partial mesh to remain within the current boundaries for switch count, ISL hop counts, and ISL loading. This design has proved to be very useful and robust for configurations of five to eight switches and can be extended further by adding switches at the center of the partial-mesh fabric.

Both the managed switch (where zoning is activated) and the principal switch should be at the logical center of the fabric for best fabric response times.

Benefits

Partial mesh designs offer extensive access to both local switch storage and single-hop storage. A partial mesh also extends the accessibility and provides many unique paths to the storage. Increasing accessibility while maintaining the same level of robustness is a design goal for every topology. Moving from many small, isolated full mesh fabrics into fewer partial mesh fabrics also simplifies the management paradigm by moving to fewer individual zone sets for the managed environment.

Moving to a partial mesh from two or more separate full mesh fabrics is a simple logical progression. Many things will remain the same and only those items that are heavily shared across the new ISL boundaries will have to be migrated toward the center of the new fabric.

Partial meshes also offer a simple progression into a core/edge design. If you look at the center of the partial mesh as the core, you can create the new infrastructure by simply removing some of the ISLs at the outer edges of the fabric. Hosts would then be physically relocated to the edges and storage would be relocated at the core switches.

Figure 22 shows a sample six-switch partial mesh. Removing the two ISLs, as illustrated, could start the migration from a partial mesh to a core/edge fabric.

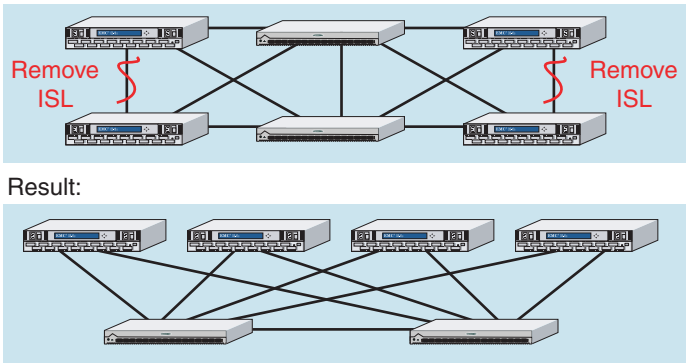


Figure 22 Partial mesh migration to core/edge fabric

Limitations

Increasing the size of the fabric always increases the dependencies within the fabric. This does not cause a problem, but it does increase the complexity of troubleshooting and the impact on unrelated processes during a fabric event.

Without rules on placement of host and storage, traffic management in a partial mesh fabric can soon become cumbersome and ISLs can become overloaded due to excessive traffic aggregation across the individual physical switch tiers. Technology advances, like the introduction of 2 Gb/s switch functionality, may be unrealized because of poor placement of these new resources or ISL placement between these new resources.

Connectivity tier fabric

Table 7 on page 65 shows the subjective rating for the connectivity tier.

Table 7 Subjective rating for connectivity tier

	Most <----------> Least				
Attribute	5	4	3	2	1
Accessibility					
Availability					
Consolidation					
Flexibility					
Scalability					
Security					
Supportability					

Each of these attributes are discussed in more detail in the "Fabric design considerations" section beginning on page 49.

Connectivity tier fabrics are based on the need to increase the accessibility between hosts and storage, and the inability of the switches at the core to support the core/edge model because of a limited number of ports per switch. The connectivity tier model creates a core of switches that are completely populated with ISLs.

The connectivity tier fabric is a three-tier physical and a three-tier logical fabric. This means that there are three switches from one side of the fabric to the other and that host to storage access traverses three switches. As for all designs, to increase the performance of the fabric management and event handling, the principal switch should be placed at the connectivity tier.

Figure 23 provides an example of a connectivity tier fabric.

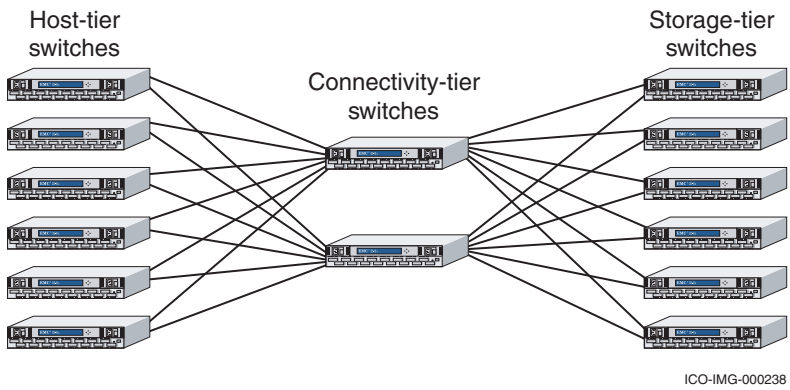


Figure 23 Connectivity tier fabric example

The tiers are:

- Host tier — The switches that are used to attach host HBAs into the fabric. Each host-tier switch should be configured using the layout recommendations for both ISLs and host attachments that are described under "Layout management" on page 67.
- Storage tier — The switches that are used to attach storage ports (disks, tapes, and so on) into the fabric. Each storage-tier switch should be configured using the layout recommendations for both ISLs and host attachments that are described under "Layout management" on page 67.
- Connectivity tier — The switches that are dedicated to aggregating and distributing the ISL traffic between storage-tier and host-tier switches. The fabric size is based on the number of ports available at this layer, as well as the individual switches' bandwidth capabilities.

  For best performance and availability, the number of ISLs coming from the host tier should equal the number of ISLs going to the storage tier. This will help to limit the chances of future traffic bottlenecks that could occur in the connectivity tier. This topology also lends itself to the placement of the principal switch in the connectivity tier, which promotes the propagation of the Class F traffic across the entire fabric.

  Zoning changes should also be initiated from the connectivity tier to facilitate the propagation of new zone sets across the fabric.

Benefits

This model allows you to extend the usage of your switches with fewer ports and increase the capacity of your fabric by aggregating bandwidth from the storage and fanning it out to the hosts through the connectivity layer.

Limitations

Without great attention to detail during the placement of component and assignment of storage access, the connectivity tier model can create severe traffic bottlenecks and eventually result in a fabric with a few overloaded ISLs, while others remain under-utilized. Since these storage switches have a limited amount of resources to handle the traffic aggregated from the host tier, you may see longer latencies and lower performance with an unbalanced layout.

The connectivity tier model also offers only two-hop access to all storage. Straying from the original layout model would impact the

accessibility benefits gained by moving to this model. Because of the dependencies this model has on the connectivity tier, losing a switch in the connectivity tier would have a large impact on the fabric performance.

Since connectivity tier switches are used only for ISL connectivity, your user port-to-ISL port ratio is lowered. Lowering the user port ratio increases the average cost associated with attaching storage and hosts into the fabric.

Nonstandard topologies

EMC does not recommend ring, string, or tree topologies because of their inherent flaws in availability, accessibility, and scalability. Single failures of components in these fabrics can result in switch isolation, resource inaccessibility, and poor performance. While these topologies can be configured, EMC highly recommends moving away from them and into a more robust and resilient design.

Layout management

Manual and purposeful managing of the layout of the individual fabric devices and ISLs can produce fabrics that are more stable, more available, less prone to latency, and less susceptible to traffic congestion. Managing traffic patterns also provides a set of rules for the systematic addition of components and promotes better ISL usage and planning, highest N_Port (storage and server) connectivity, better asset management, and apportioning analysis. You may not be able to implement everything that is being proposed in this section, but the following recommendations are listed in the priority that they should be implemented.

General recommendations

1. Map volumes across multiple storage ports for availability.
2. When mapping storage ports that share paths to the same devices, or HBAs in the same host that share paths to the same devices:
    a. If mirrored fabrics are being used, attach the ports to different fabrics.
    b. If mirrored fabrics are not being used, attach the ports to different switches in the same fabric.

c. If the ports must be attached to the same Fibre Channel director, attach the ports to different port cards.

d. If the ports must be attached to the same departmental switch, attach the ports different ASICs on the same switch.

When mapping ISLs in your fabric:

1. Connect each switch to more than one other switch in the fabric.
2. Between any two switches there should be at least two physical connections for redundancy. Additional ISL requirements should be based on the actual/estimated performance data.
3. Choose one of the following:
   - If hardware trunking (such as with Brocade) is *not* being used:
     - For directors, connect ISLs to the same switch across different port cards.
     - For departmental switches, connect ISLs to different switch ASICs.
   - If hardware trunking *is* being used (refer to "Trunking" on page 207), you should also consider the redundancy benefits of creating two separate trunks as opposed to one larger one when planning trunking implementation.

Switch interoperability

Switch *interoperability* refers to the formation of a fabric using switches from multiple vendors. E-Lab Navigator contains fabric guidelines, the vendor switch models, and code versions that are supported in a heterogeneous fabric. For more information and implementation examples, refer to "Interoperability" on page 302.

Multisite fabrics

This section contains the following information related to multisite fabrics:

Catalysts for multisite fabrics

Resource sharing

Connecting two or more sites into a single fabric allows you to create a sharable resources pool that includes all of the free resources available at each one of the connected sites. This is especially useful when looking for free disk storage or building a disaster recovery location. Sharing resources like hosts and storage among the different sites can increase your resources usefulness and overall company productivity.

Disaster recovery

Once connectivity is established between the sites, you might choose to create remote data facilities using the Symmetrix RDF or VNX series or CLARiiON mirroring applications. More and more customers are moving to a centralized SAN backup architecture that allows them to locate their tape storage resources in one physical location.

Backup centralization

Creating a centralized backup location, where both the trained backup administrators and the backup media resources are located, can increase the duty cycles of your backup media, as well as increase the efficiency of your personnel resources. Centrally locating the personnel fosters communication, facilitating access to, and transfer of, skills among the group.

Digital vaulting

Digital vaulting is the procedure of performing daily backups over an extended-distance link to a remote data facility. The extent of your digital vaulting can included both the original backup and a clone, or it might include only the tape clone. Administrators might even choose to perform a backup to a disk that is part of an RDF group. This disk information would then be vaulted to a remote site's R2 device.

Digital vaulting can employ any combination of RDF, BCVs, or other backup methodology that best fits your needs for backup and recovery. No matter which procedure is used, the time and expense savings associated with switching to a digital vaulting paradigm should be evaluated in your business model.

Creating multisite fabrics

Fabrics can be created that can expand beyond the physical constraints of a room, a floor, a building or a site. Multimode Fibre Channel links support devices over 500 meters apart at a 1 Gb/s speed. When the speed is increased to 2 Gb/s, the distance is reduced to 250–300 meters. If your needs go beyond this distance, you may

decide to use either longwave optics or fabric-extending DWDM technology. E-Lab Navigator lists the support distances for longwave optics and the list of currently supported DWDM vendors.

EMC recommends using longwave and DWDM connectivity to extend the length of an ISL. Extending the distance of ISLs gives you the greatest efficiency on your long-distance investment by allowing you to share the ISL link among multiple storage devices and multiple hosts devices on either side of the distance-extension device.

Sharing the ISL between multiple devices maximizes the usage of the link, and increases the usefulness of the distance extension components.

Whenever extending the distance of a device or an ISL, you should be aware of the buffer-to-buffer credit requirements to maintain data transmission performance over the link.

Multisite example: Application/backup and RDF

Mirrored fabric with DWDM across the mirrors

Figure 24 shows how two sites may be connected by DWDM equipment using a portion of their mirrored environment at both sites.

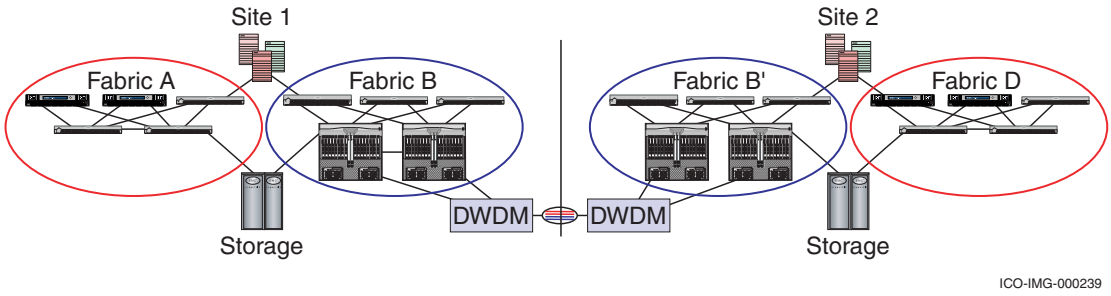


Figure 24 DWDM across mirrored fabric

While linking the mirrors on both sides creates a single larger fabric from fabrics B and B', it does not require additional switch resources at either site. This topology would allow direct sharing of storage resources between both sites, as well as RDF traffic between the two sites. Data traversing from site 1 to site 2 travels across the back end of fabric B to the back end of fabric B'. This will eliminate any impact on the subscriber traffic at one site during the sharing of resources from the other site.

ISLs could also be added from fabric A to fabric D to create another multisite fabric (A+D). Leaving them separated limits dependencies

on the long-distance links between sites. An event on the long-distance links that causes a flux in the system would affect only fabric B+B'. Fabrics A and D would be isolated from the event.

RDF extended fabrics

Figure 25 shows how to configure a mirrored fabric environment and reduce the dependencies so that only the RDF traffic traverses the link. For accessibility and performance, EMC recommends that Fibre Channel switches be placed on both sides of the DWDM equipment.

This ensures that there are enough BB_Credits to handle the bandwidth over the required distance and that there is enough bandwidth aggregation to efficiently use the long distance Fibre Channel links.

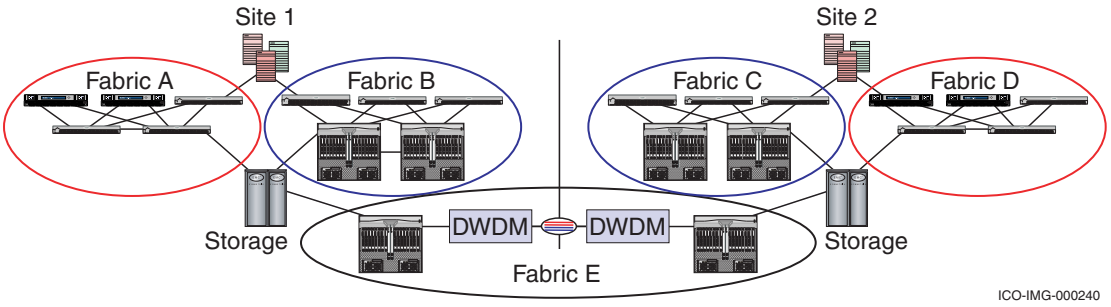


Figure 25 RDF extended fabric

Tape connectivity

EMC supports attaching tape drives into the fabric using Fibre Channel/SCSI bridges, FC-AL/FC-SW bridges, FL_Port-capable switches, or native FC-SW-capable tape drives.

Though tape drives may be placed anywhere in the fabric, EMC has developed some best practices resulting from qualification efforts and customer testimonials. These recommendations are based on optimizing the ability to share the tape drives with multiple hosts, as well as ways to minimize the impact of tape backup traffic on the application traffic. This section will further discuss:

- "Tape drive-sharing model" on page 72
- "Traffic reduction model" on page 73

With any topology that you choose, EMC recommends that the design be mirrored and balanced whenever possible. Since tape

drives are supported as single attached devices, you should consider spreading your tape drives across both sides of the mirrored fabric for highest availability of the backup solution. Your backup software solution should be evaluated for the best way to create pools of devices that can include tape drives from both fabrics.

PowerPath scripting should also be evaluated to investigate how setting an HBA that is shared to both tape and disk into standby mode may boost the performance of your backup.

Setting a path to standby still allows PowerPath to use it in case of a path failure, but would not use it during normal conditions for disk traffic. This procedure is especially useful for hosts that do not have enough available slots for a dedicated tape (backup) HBA.

Tape drive-sharing model

To promote sharing of any devices, you would place those devices at the center (core) of the fabric. This would allow all hosts on the fabric to have equal, deterministic access, equal bandwidth performance, and equal latency to the shared device. The farther a shared device is from a host, the more traffic aggregation that host must contend with to reach and access the device. If only a few media servers are solely responsible for writing the data to disk, these media servers would be placed on the same switch as the tape drives, or as close to the tape drives' switches as possible.

Figure 26 shows an example of a tape drive-sharing model of a fabric where a core-to-edge fabric topology has been deployed.

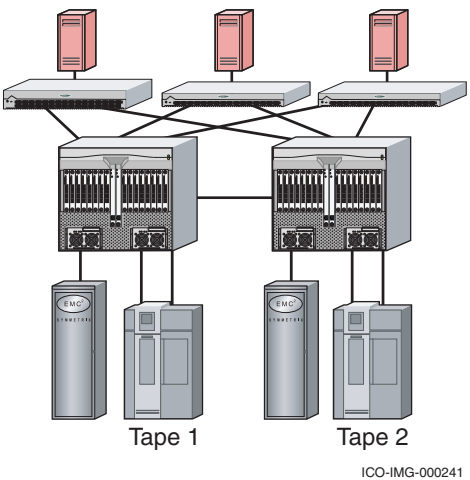


Figure 26 Tape drive-sharing model

With the advancement of server-free backups, EMC recommends that the copy engine, storage ports, and tape drives be located on the same switch or at the back end of the fabric.

You can construct a front-end/back-end design, as done in Figure 26 on page 72, so that backup data will traverse only the core-to-core ISLs in a server-free backup environment.

Traffic reduction model

If tape devices are highly dedicated and seldom shared across servers, placing them on the same switch as a tape-capable application server or on the same switch as the media servers can help reduce the traffic associated with backups.

Data must be:

1. Read from the disk storage at the core.
2. Pulled across the core-to-edge ISL.
3. Brought into the backup (media) server.
4. Written on one of the tape-accessible HBAs.
5. Sent back over the ISL from the edge to the core.
6. Written to tape.

If you were concerned only with the amount of ISL traffic, you can see that the data has traversed the ISLs twice. Though this is a powerful fabric design, you can see how the number of traversals could grow as you move to higher-tier fabrics like the partial-mesh fabrics or connectivity-tier fabric topologies.

In Figure 27 on page 74, you have moved the tape devices out toward the edges on the same switches as the tape-accessible hosts (application servers or media servers). In this model you have to traverse the ISLs only once during the operation where you are initially reading from the disk array.

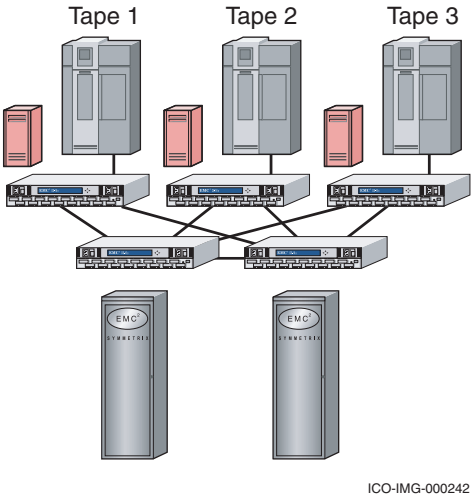


Figure 27 Traffic-reduction model

Sharing tape storage in this model between hosts becomes more costly because you would have to make three ISL transitions to use a tape that is not connected to the local switch. This design would also not reap the full benefit of an environment that was employing a sever-free backup solution.

General fabric design recommendations

Through testing and customer experience, EMC has compiled a list of general fabric design recommendations. These recommendations can be used no matter which switch vendor you are using or which topology and protection schemes you have employed at your site. Recommendations are discussed for the following areas in this section:

- "Physical topology hop count" on page 75
- "Maximum switch count per fabric" on page 75
- "Maximum number of Nx_Ports in the fabric" on page 76
- "Unique Domain IDs" on page 78
- "Fabric mode" on page 79
- "ISLs in the fabric" on page 79
- "Principal switch placement" on page 83

- "Zoning" on page 83
- "Port/cable sparing" on page 84

Consult the E-Lab Navigator for the most up-to-date information.

Physical topology hop count

For all EMC-qualified switches, EMC recommends that you build your initial fabrics so that all components are within three hops. If you were to build a larger fabric, you may find that your fabric is unable to maintain the fabric topology or that it drops frames after a failure of one or more components.

The failure of one or more components may change the topology of the fabric in such a way that you are exceeding the time a frame is allowed to stay active within the fabric. Changing the time-out values can cause undue congestion in the fabric if frames were to be allowed to remain active for long periods of time without reaching their destination. Fabric recovery and change response times are also based on these values and altering these values without a full understanding of the consequence could result in unexpected fabric behavior.

One such example of a fabric change would be a zone set merge due to the attachment of two operational fabrics. Once the fabric parameters and link parameters were verified and the link came up, the zone set merge would propagate from the initial point of connection through the adjacent switches. Once an adjacent switch successfully merged the zone set, it would then propagate the merge request to its adjacent switches. The time that it would take to do this increases as the distance from the initial connection point increases.

If the placement of components around the fabric becomes unmanaged, the traffic patterns will become nondeterministic and as the size of the topology increases, the configuration may become more susceptive to bandwidth congestion problems. Maintaining a densely-configured fabric of three hops helps prevent nondeterministic traffic and traffic congestion issues.

As technologies and software advances are made, EMC will continue to extend the solution environment envelope. E-Lab Navigator describes changes to these recommendations.

Maximum switch count per fabric

As with the number of hops in a fabric, the number of switches in the fabric also affects the operation and timing of fabric responses. Fibre Channel switches communicate among themselves to provide distributed services for zone propagation, name resolution, and switch identification. As the number of switches increases, so does

the amount of data that is propagated and the amount of components involved in the negotiation and synchronization of this data.

Since the effects of fabric size are very important to the continued reliability of the fabric, EMC continually spends a great deal of effort identifying and progressing the current fabric limits.

Fabric size increases are qualified in the same detailed fashion as would be the introduction of a new switch model or new firmware revision. Increases in the number of switches in the fabric are tested within the specific topologies listed in this document to continually validate and improve on our design practices.

Currently, EMC recommends that a maximum of 31 switches be used in a single fabric. E-Lab Navigator describes changes to these recommendations as well as the switch types capable of this fabric size.

Maximum number of Nx_Ports in the fabric

The single largest limiting factor in building large fabrics is the number of Nx_Ports that will be attached. This is due to the amount of information that must be passed between all distributed name servers in the fabric each time a new Nx_Port is added or removed.

One or two ports entering or leaving the fabric usually has a very small impact, but when you start adding and removing an almost fully-populated director to the fabric, the impact can be enormous.

Figure 28 on page 77 and Figure 29 on page 77 illustrate the impact to name server response time during a Zone Set activation and a Domain Add. The testing was being performed as a part of the validation work done to verify that a newer version of firmware from one of EMC's switch vendors could handle up to 2048 ports in the same fabric. The zone set activation chart (Figure 28) shows that on the older version of firmware, the name server would take up to 6 seconds to respond to NS Queries when a zone set was activated and 1530 Nx_Ports were in the fabric. This same chart also shows that the response time dropped to less than a second with almost 2000 Nx_Ports.

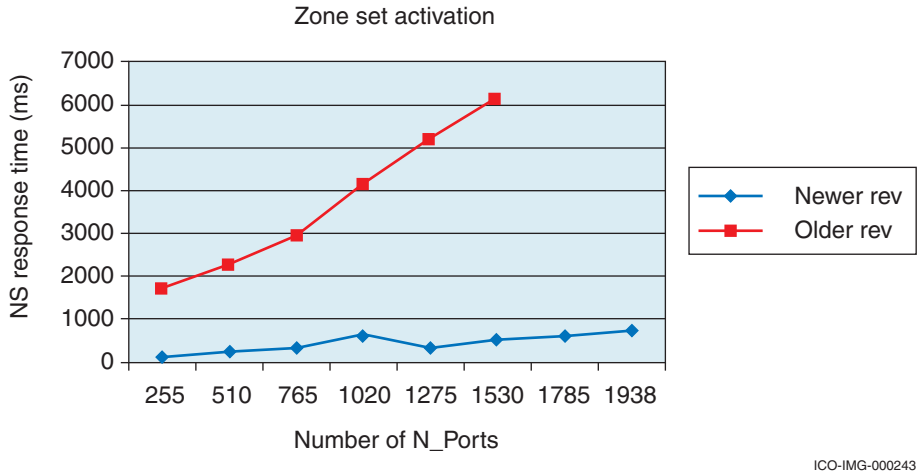


Figure 28 **Zone set activation**

The Domain add chart (Figure 29) shows that on the older version of firmware, the name server response time was 10 seconds when adding the 512th through 765 devices. The same testing on the newer version of firmware shows that up to 2000 ports can be added and response time remains less than 10 seconds.

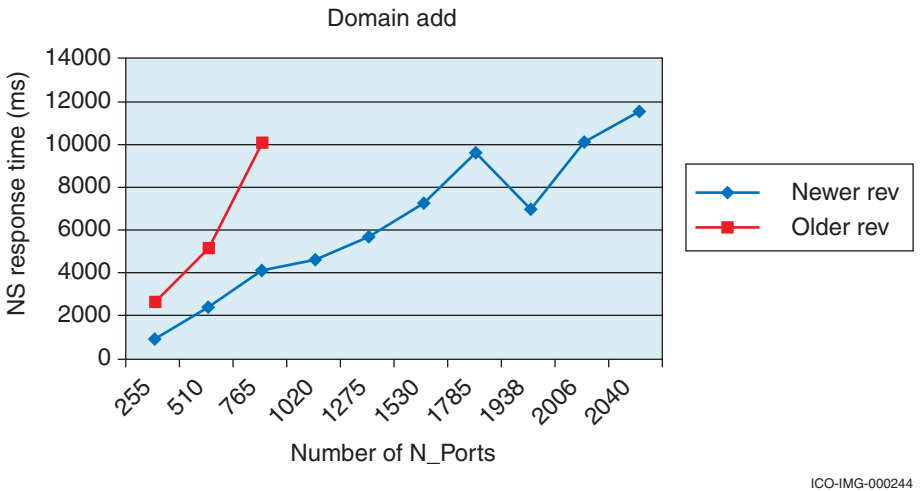


Figure 29 **Domain add**

Later versions of firmware from this vendor also showed improvements that allowed us to support up to 2048 Nx_Ports in the same fabric.

You may be asking why E-Lab cares about Name Server response time. Two answers spring immediately to mind.

- Name server response time impacts the amount of time it takes for two Nx_Ports to establish or re-establish connectivity through the fabric. This establishment or re-establishment of connectivity is also known as *discovery*. While it is not normal for host and storage pairs to be utilizing the name server to perform discovery during normal operation, it is possible that discovery will be invoked as a part of some error recovery sequence during a domain add or zone set activation. If this were to happen and the name server was unresponsive for long periods of time, the outcome would be unpredictable.
- Name server response time is one way to measure how busy a fabric is. The more busy a fabric is during a simple zone set activation or domain add, the higher the likelihood that the fabric will have a more difficult time recovering from error scenarios that are not as straight forward. The amount of time it would take to perform the recovery would also be impacted.

Unique Domain IDs

EMC recommends that you always attempt to give each switch a unique preferred Domain ID, whether it is targeted for an existing fabric or a new fabric. (The exception to this rule would be if you were swapping a new switch for a failed switch.) This provides the best opportunity to merge a switch or fabric into another fabric with the least amount of manual intervention.

Since Domain IDs become part of an HBA or storage array port's fabric address, and data is transferred to and from devices based on their address, Domain IDs cannot be changed without first bringing the switch to an offline state. Every switch that is currently online will have been assigned a Domain ID from the principal switch. If the fabric contains only one switch, that switch is the principal switch for that fabric. Domain ID negotiation done during a fabric merge will not change the current Domain ID when the switch is connected to the fabric while online.

During the exchange of parameters following a merger event, if the switches identify a domain conflict, they will not merge and the ISLs connecting them will remain segmented. Taking a switch that has segmented from the fabric due to a Domain ID conflict offline and

then bringing it back online allows the switch to renegotiate for a new unique Domain ID. It should be noted that taking a switch offline terminates communication between the devices that are on that switch. EMC recommends that if a conflict does exist that you change the preferred Domain ID of the switch yourself so that you can ensure that it is unique in your entire environment and that it will always attempt to use the same ID, even after future merges and fabric changes.

Fabric mode

Fabric mode refers to the switch code that is utilized to operate the switch-to-switch communication and distributed processes involved with building and maintaining a running fabric.

Originally, each vendor created its firmware with a slightly different interpretation of the Fibre Channel standard. This meant that it would not be able to operate natively in a heterogeneous switch vendor environment.

Changes were made and a new *Open Fabric* mode was created that results in a common interpretation and implementation of these required switch-to-switch communications. To operate in a heterogeneous switch environment, all switches in the environment must be set to Open Fabric mode.

Changes to the fabric mode of a switch that allow it to operate in a single vendor fabric or a heterogeneous vendor fabric changes not only the fabric mode, but also the switch address. Both of these changes mandate that the switch first be set offline before the change is made. Once the change is made the switch may be brought back to an online state.

Since the change of the mode is an offline change, you should consider how to set the mode during the initial installation of a new switch or new fabric.

E-Lab Navigator contains more information on the specific supported code combinations between vendor switches.

ISLs in the fabric

Interswitch links (ISLs) are used to transfer host-to-storage data, as well as fabric management traffic from one switch to another. When planning the quantity and locations of ISLs around the fabric, one should consider the effects of routing, redundancy requirements, ISL load balancing, and trunking. Following a brief discussion of each of these, ISL design guidelines are provided.

Intra-fabric routing

Not to be confused with inter-fabric routing, intra-fabric routing describes the process used to route frames from an ingress port (entry point) to egress port (exit point). Frames are routed across the fabric using an algorithm called Fibre-Shortest-Path-First (FSPF) routing. Each ISL in the fabric is assigned a cost based on the link speed it is currently operating at. FSPF calculates the shortest distance between an ingress and egress port by finding the path with the lowest cost. The path with the lowest cost will typically be the one(s) with least number of hops, but this may not always be the case when mixing different line rates (4 Gb/s and 1 Gb/s especially) in the same fabric.

EMC strongly recommends that you construct your fabric to have multiple equal, lowest-cost, shortest-path routes between any combination of host and storage. This means that you may have two ISLs between every switch in the fabric or you may have single links between switches, but would still have multiple equal-cost/length paths that travel through different switch combinations in the fabric.

Routes between storage and hosts that are not the shortest, lowest-cost path will not be used unless there is an event in the fabric that causes them to become the shortest, lowest-cost path.

Routing tables on each switch are updated and recalculated during events that change the status of links in the system. The calculation of routes and its ability to perform this function in a timely fashion is important for the stability of the fabric. For this reason, as well as for the fact that for every ISL configured, two ports are lost to attaching storage and hosts, EMC recommends using reasonable limits on the number of ISLs in a fabric. Since only the lowest cost/shortest-path routes will be used in the routing scheme, many ISLs may remain unused while others may be approaching peak utilization as long as there are no events in the fabric. For a true number of required ISLs, you should continually examine your ISL utilization and identify your level of actual protection from fabric events. You may be able to identify how changes in your current topology could enhance your performance as well as your path redundancy.

Routes are assigned to devices for each direction of the communication and the route one way may differ from the return route. The routes are assigned based on a round-robin approach that is initiated as the device is logged in to the fabric. These routes are static for as long as the device is logged in or routes do not have to be recalculated due do a fabric event.

Redundancy requirements

Redundancy is added to the fabric to protect your environment from possible events and failures that may occur. The amount of redundancy that you choose to add to the fabric is a decision based on your business model and the amount of resources you can spare for increased availability.

EMC recommends a minimum of two ISLs between any two switches for redundancy. This also allows a certain level of traffic balancing across the ISLs and maintains a same-cost route in the fabric if one of the links fails.

EMC also recommends connecting each switch to more than one switch for redundancy. Having each switch connected to more than one switch in large configurations ensures multiple paths to the edge switches if one of the intermediate switches or paths to those switches fails.

If you are limited on the number of ISLs you can place in the fabric, EMC recommends configuring a single ISL from the originating switch to multiple switches if both paths from the hosts to its storage could be counted as equal, shortest-path, lowest-cost paths.

These recommendations can be used separately or combined for even higher availability. The ultimate goal is to provide each host multiple primary paths to its storage and then a number of secondary paths if failures were to occur. ISL utilization should always be monitored to identify unused, under-used, or over-utilized ISLs. Unused ISLs could become candidates for removal if they do not represent the only secondary path a host would have to its storage in the event of a switch or ISL failure.

E-Lab Navigator lists the ISL limits for individual switch vendors.

ISL load planning

There is no substitute for real traffic analysis and identification of burst patterns in your information model; however, there is a set of initial design rules that can be used to implement your initial fabric design.

When designing a new fabric from the ground up, use the 6:1 rule to get a general feeling for the number of ISLs that will be required between two domains. The rule states that for every six Nx_Ports that will be routed between two domains, there should be one ISL.

For another approach, consider the following.

Performance on a storage port is highly dependent on the number of I/O requests per second and the size of the I/O in the request. This will vary from customer to customer and from process to process. It has been found that a reasonable design guideline can be estimated when using the EMC fan-out recommendations for storage ports. EMC currently recommends two Symmetrix Fibre Channel director groups per ISL for initial fabric planning. (A Fibre Channel director group is the set of host HBAs that are communicating with a particular storage port.) This estimate uses as its basis the 1 Gb/s dual port/dual processor boards. Using the connectivity-enhanced four port/dual processor boards, the recommendation changes to four Fibre Channel director groups per ISL for initial fabric planning.

As an example, if the fan-out of a storage port is 12:1 (12 HBAs to one storage port), you would plan to allow the traffic from two storage ports to traverse from one switch to another over a single ISL. For redundancy reasons, you should add one ISL for increased availability. When considering SRDF ports in your design, the initial design estimates are similar. EMC recommends that for initial design purposed you plan to use two local SRDF ports per ISL.

After the design has been implemented, all ISLs should be monitored and readjusted based on the actual results of the fabric over time. Data bandwidth requirements continually grow, so they should be monitored and tracked with care. Regular monitoring will assist in planning future needs with greater confidence and accuracy. It should also allow the lead time required to better plan for the acquisition and implementation requirements of the new equipment.

Planning ISLs for tape backup is slightly more complex. Because individual tape drivers do not have the load balancing, path failover, or random simultaneous host access potential that disks have, you need to consider only their individual access potential and your redundancy concerns. The redundancy concerns can be satisfied by spreading out our total complement of tape drives across multiple switches. This will ensure that you do not have all of your tape drives traversing over the same single link.

To gauge the additional bandwidth requirements for tape backup, you can use the bandwidth specifications for the tape using the vendor's maximum compression numbers. This will give you the most conservative effort for allowing for the greatest possible bandwidth requirements. Once you have this number, you can simply take 80 percent of the ISL bandwidth and divide it by the bandwidth of the tape with maximum compression. This will give

you the number of tapes that can be streaming across the ISL at one time.

Combining the outcomes of the SRDF, storage port, and tape drive bandwidth numbers will provide a high-confidence estimate for the quantity of ISLs required between the switch tiers.

E-Lab Navigator contains the latest information on the number of host HBAs that may be zoned to a single storage port by OS platform.

See "Determining customer requirements" on page 91 for more information.

For ideas on how to gauge your current traffic patterns and traffic requirement, you can examine the bandwidth reporting available in EMC ControlCenter and Connectrix Manager.

ISLs would then be added based on your plans for a balanced fabric and your actual bandwidth requirements. Loading plans should be such that the no ISL in the planning stage is scheduled to exceed 80 percent of its actual bandwidth potential.

Designing for greater bandwidth could result in an inability to compensate for bursty traffic or extrapolation errors that may have occurred during the planning stages.

Principal switch placement

Domain ID negotiation is governed by the entity in the fabric called the *principal switch*. Principal switch selection is made based on the switch with a combination of the highest principal switch priority and the lowest World Wide Name.

Proper placement of the principal switch in the fabric can lead to short negotiation times, resulting in the fabric returning to a normal state in an shorter amount of time. For these reasons, EMC recommends that the switch at the logical center of the fabric be made the principal switch. This would normally mean the switch with both the least amount of hops to the farthest extent of the fabric and/or the switch that has connections to the most other switches in the fabric.

These two placement strategies will help to ensure that your principal switch access times are as quick as they can be.

Zoning

To an end user, zoning is one of the most critical concepts to fully understand, yet it seems to be the most widely misunderstood. Zoning is a generic term that is commonly used to describe either the

process of grouping ports together so that they may access each other over the fabric *or* the effective zoning configuration on the fabric.

The process of zoning consists of identifying host and storage pairs, adding them to a zone, adding this new zone to a zone set, and then activating the zone set onto the fabric where the host and storage pair resides. Once the zone set is activated, the host and storage pair can communicate with each other.

General guidelines for zoning

- Use a consistent and detailed zone and zone set naming convention.
- Use WWPN-based zoning and not port zoning.
- Configure each zone so that it only contains one initiator.
- Practice good SAN hygiene and frequently check for "dead" zones.

Port/cable sparing

Sparing needs are associated with the resources you are leaving unused as well as the resources that can be removed and reused based on your business's processed prioritization model. Only you can decide what can be reused for another process based on your business prioritization model. Therefore this section will focus on what issues need to be considered when calculating for true spares, as well as which resources should be spared in a fabric environment.

Cable sparing

The data center is sometimes a very busy and dynamic environment. New equipment implementation, troubleshooting techniques, and everyday data center activities can place the Fibre Channel cabling at some risk of damage. Whether you are using patch panels for easier cable change management, or raised floors and cable trays for cable protection and concealment, you still may be at risk. Cable shielding and connectors can be compromised if they are under some continued weight-related stress associated with how they are attached and supported on the patch panel or Fibre Channel component. Over time, the light loss associated with the connections could increase due to these stresses. Some cables have been damaged from a heavy floor tile being dropped. For these reasons, EMC recommends that you always have spare cables on hand at your site.

When running trunks from a Fibre Channel device, EMC recommends that the links to this device be split among several different, smaller trunks, and that those trunks are run under a separate set of floor tiles some distance from each other. This is the

same analogy used when DWDM rings are established; both sides of the ring should never be run through the same location. This analogy can be brought into the data center and increase the resilience here as well.

This process is completed by maintaining a cable plant diagram that lists the cable trunks, their role in the data center, their start and end points, the paths that they travel, and their position at the end devices. Maintenance of this plan can greatly reduce the time needed to troubleshoot suspected cable problems. It will also help you redeploy cable trunks as equipment assets are shifted. Cable trunk spares may be used not only for the devices that they are currently servicing, but also for neighboring devices when their trunk spares have been exhausted. This cross-utilization allows for a more efficient sparing model at the site.

EMC recommends that you run extra lines with each trunk. These lines can be used for *fast starts* associated with getting a new system up and running as soon as possible, as well as replacements for suspected troublesome cables. The number of cables you spare in trunks or lose depends on how dynamic your environment is and how subject it is to stresses.

ISL port sparing

Data and performance requirements will continue to grow as time goes by. To manage that growth you may need to add ISLs between adjacent switches. To do this, you would first have had to reserve spare ports on both switches to connect the new ISL. Without sparing the ports in the initial design, you would first have to migrate components off of the selected switches onto other switches. This migration would cause at least a momentary dip in performance and, depending on the redundancy in the fabric design, could cause a temporarily delay of services. If care is not taken, the resultant new data path may be longer and more congested than the original, thus causing another congestion issue that will need to resolved.

While this section talked about growth of an existing fabric, there is also the growth associated with the addition of new switches and the merging of existing fabrics for the purpose of increasing our storage accessibility. When you are attempting to identify the quantity of sparing ports for both growth as well as for expansion, you need to consider both the location and direction of these new ISLs. Each topology design will have its own methods for adding ISLs for added bandwidth and merging fabrics.

Sparing for failures

Failures of switch components, while rare, can occur in your fabric. The failure of a port card in a director would force you to find free ports for most, if not all, of the attached ports on the failed card. Since failures are a temporary state and you are always expected to have the failed component replaced, you may decide to move only those items that are necessary in the short term. To provide for a card, port, link, or optic failure, you must first understand how any one failure of any one component could effect your business continuance. Depending on your current protection schemes (for example: mirroring fabrics, PowerPath, or sparing models), you can identify which resources need protection and preservation during an outage. Any event in the fabric will do its best to work around the problem automatically. For this reason, you need to consider sparing as a means to protect the relative bandwidth performance of particular applications and lines of business. The planning of such an event, as well as its recovery procedure, is just as important as the reservation of resources to handle these unexpected events.

EMC recommends that if you are unprotected, or weakly protected, you should spare extra ports in your fabric for failure events. You may find the following useful when considering where and how many ports to spare in your environment:

- Sparing per switch — This allows you to avoid any logistics associated with moving things to a new Domain ID or recabling to reach the other switch. It may also allow you to avoid any issues associated with the rebalancing of the bandwidth load because the traffic patterns have changed.
- Sparing per fabric tier — This may allow you to avoid any logistics associated with rebalancing the bandwidth load because the traffic patterns have changed. This model may allow you to spare less if you are able to redeploy your cabling to make these changes.
- Sparing per fabric — This allows you more flexibility and could reduce the overall spare ports required, but would cause you to possibly rebalance the fabric traffic. If this is extended to include the sparing for both sides of a mirrored fabric, you would have to include both rezoning and migration of components currently attached to the other side of the fabric. Since the interdependence of the storage components may be very complex, EMC recommends that, whenever possible, you do not create a sparing model that includes the migration of components to other fabrics.

- Sparing per site — EMC strongly recommends that you always reserve some level of sparing for each individual site within your business. If you are severely limited by the number of spares that you can reserve, you may want to explore a recovery plan that calls for a merge of smaller environments that shares the spares among the entire environment. The detailed planning of such an event is the most important step in its successful execution.

Cable/fiber types and supported distances

You may use any cable/fiber that meets or exceeds local regulations for the protocol to be used, as long as it does not exceed the following:

- Maxmimum support length
- Loss budget allowed

For distances exceeding the listed coverages, refer to the *Extended Distance Technologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab, or refer to the *EMC Support Matrix*.

Table 8 provides more information on multimode media maximum distances.

Table 8 Multimode media maximum distances

Protocol	Transceiver type	Speed	Multimode media maximum distance			
			62.5µm/200MHz* km (OM1) [62.5 micron]	50µm/500 MHz* km (OM2) [50 micron]	50µm/2000 MHz* km (OM3) [50 micron]	50µm /47000 MHz* km (OM4) [50 micron]
FC	SW	2 Gb/s	150m	300m	500m	500m *
		4 Gb/s	70m	150m	380m	400m
		8 Gb/s	21m	50m	150m	190m
		10 Gb/s	33m	82m	300m	550m
		16 Gb/s	15m	35m	100m	125m
GbE	SW	1 Gb	300m	550m	1000m	1000m *
		10 Gb	33m	82m	300m	550m
		40 Gb	N/A	N/A	100m	125m

* Denotes at least this distance. No documented distance is available at this time.

Table 9 lists single-mode media maximum distances.

Table 9 Single-mode media maximum distances

Protocol	Transceiver type	Speed	Single-mode media maximum distance
			9µm [9 micron]
FC	**LW**	2 Gb/s	10 km
		4 Gb/s	10 km
		8 Gb/s	10 km
		10 Gb/s	10 km
		16 Gb/s	10 km
GbE	**LW**	1 Gb	10 km
		10 Gb	10k m
		40 Gb	10 km
		100 Gb	10k m

Table 10 lists Twinax distances.

Table 10 Twinax distances

Protocol	Transceiver type	Speed	Twinax
GbE	**SW**	1 Gb	5m
		10 Gb	5m
		40 Gb	5m
		100 Gb	5m

Resources

For additional details please refer to the following sources and standards:

FC-PI-5 at http://www.T11.org

10G FC at http://www.T11.org

IEEE Standards Association, 802.3ae-2002, at http://standards.ieee.org

Fibre Channel over Ethernet TechBook on the E-Lab Interoperability Navigator, **PDFs and Guides** tab

IBM Storage Network presentation at http://www.ibm.com

OM4 Fiber – The Next Generation of Multimode presentation at http://www.fols.org/documents/OM4Final.pdf

IEE P802.3ba D3.0 40 Gb/s and 100 Gb/s Ethernet comments, Jan10 P8023ba-D30 Proposed Responses by ID, at http://ieee802.org

100 Gb Ethernet information at http://en.wikipedia.org/wiki/100_Gigabit_Ethernet

Cisco SFP Optics For Gigabit Ethernet Applications product data sheet at http://www.cisco.com

Choosing the Right Multimode Fiber for Data Communications white paper at http://www.fols.org/fols_library/white_papers

Determining customer requirements

Note: Although this section assumes that the reader will be deploying a completely new SAN, the user adding capacity may find some useful information in this section, as well.

Once you have an idea of what a SAN will be used for and the physical location of each piece of equipment settled, you need to consider how many host and storage ports will be deployed initially, and how the environment is expected to grow, before you can decide on the right topology.

The following information is provided in this section:

- "Scalability", next
- "Choosing a switch type" on page 95

These are just guidelines. Your actual implementation will vary as more or less ISLs are needed between switches.

Scalability

Scalability information for full mesh with two and four ISLs and full mesh core with edge switches has been included below to help you find the right topology.

Full mesh core with two and four ISLs

In Table 11, next, and Table 12 on page 92, combinations that result in negative numbers or result in greater than 2048 available ports are grayed out and should *not* be considered for use.

Table 11 Full mesh with two ISLs

Number of ISLs between switches	2																	
	Number of ports on switch																	
Number of switches	16			24			32			64			140			256		
	#otal	# avail	%avail	#otal	# avail	%avail	#otal	# avail	%avail	#otal	# avail	%avail	#otal	# avail	%avail	#otal	# avail	%avail
1	16	**16**	100%	24	**24**	100%	32	**32**	100%	64	**64**	100%	140	**140**	100%	256	**256**	100%
2	32	**28**	88%	48	**44**	92%	64	**60**	94%	128	**124**	97%	280	**276**	99%	512	**508**	99%
4	64	**40**	63%	96	**72**	75%	128	**104**	81%	256	**232**	91%	560	**536**	96%	1024	**1000**	98%
8	128	**16**	13%	192	**80**	42%	256	**144**	56%	512	**400**	78%	1120	**1008**	90%	2048	**1936**	95%
16	256	-224	-88%	384	-96	-25%	512	**32**	6%	1024	**544**	53%	2240	1760	79%	4096	3616	88%

Table 12 Full mesh with four ISLs

Number of ISLs between switches	4																	
	Number of ports on switch																	
Number of switches	16			24			32			64			140			256		
	#total	# avail	%avail	#total	# avail	%avail	#total	# avail	%avail	#total	# avail	%avail	#total	# avail	%avail	#total	# avail	%avail
1	16	**16**	100%	24	**24**	100%	32	**32**	100%	64	**64**	100%	140	**140**	100%	256	**256**	100%
2	32	**24**	75%	48	**40**	83%	64	**56**	88%	128	**120**	94%	280	**272**	97%	512	**504**	98%
4	64	**16**	25%	96	**48**	50%	128	**80**	63%	256	**208**	81%	560	**512**	91%	1024	**976**	95%
8	128	**-96**	-75%	192	**-32**	-17%	256	**32**	13%	512	**288**	56%	1120	**896**	80%	2048	**1824**	89%
16	256	-704	-275%	384	-576	-150%	512	**-448**	-88%	1024	**64**	6%	2240	1280	57%	4096	3136	77%

The following definitions apply to these tables:

Number of switches	The number of switches in the fabric.
Number of ISLs between switches	Indicates the number of ISLs that will connect every switch to every other switch. Since this is a full mesh, all switches will connect to each other.
Number of ports on switch (i.e., 16, 24, 32, etc)	Indicates the number of ports on each switch in the fabric. It assumes all switches have the same port count.
#total	The total raw port count. As of publication, the raw port count cannot exceed 2048 in any single fabric. This number can be determined by summing the port count on each switch.
# avail	The number of ports available for Nx_Ports to attach to.
% avail	The amount of ports that were not consumed by E_Ports expressed as a percentage of the total ports. Generally, you should not use a topology that has 50% or less of the ports available.

Full mesh core with edge switches

Table 13 and Table 14 on page 93, and Table 15 on page 94 provide information for 64, 140, and 256 port core switches, respectively.

Table 13 Using 64 port core switches

Number of ports on core switches	64						
Number of ISLs between core and edge	2	Number of core switches			4		
Number of cores each edge will connect to	2	Number of ISLs between cores			2		
Number of Edge switches	Core ports	Number of ports on edge switch					
		16	24	32	64	140	256
1	**228**	12	20	28	60	136	252
2	**224**	24	40	56	120	272	504
4	**216**	48	80	112	240	544	1008
8	**200**	96	160	224	480	1088	2016
16	**168**	192	320	448	960	2176	4032
24	**136**	288	480	672	1440	3264	6048
32	**104**	384	640	896	1920	4352	8064

Table 14 Using 140 port core switches

Number of ports on core switches	140						
Number of ISLs between core and edge	2	Number of core switches			4		
Number of cores each edge will connect to	2	Number of ISLs between cores			2		
Number of Edge switches	Core ports	Number of ports on edge switch					
		16	24	32	64	140	256
1	**532**	12	20	28	60	136	252
2	**528**	24	40	56	120	272	504
4	**520**	48	80	112	240	544	1008
8	**504**	96	160	224	480	1088	2016
16	**472**	192	320	448	960	2176	4032
24	**440**	288	480	672	1440	3264	6048
32	**408**	384	640	896	1920	4352	8064

Table 15 Using 256 port core switches

Number of ports on core switches		256					
Number of ISLs between core and edge		2	Number of core switches			4	
Number of cores each edge will connect to		2	Number of ISLs between cores			2	
Number of Edge switches	Core ports	Number of ports on edge switch					
		16	24	32	64	140	256
1	**996**	12	20	28	60	136	252
2	**992**	24	40	56	120	272	504
4	**984**	48	80	112	240	544	1008
8	**968**	96	160	224	480	1088	2016
16	**936**	192	320	448	960	2176	4032
24	**904**	288	480	672	1440	3264	6048
32	**872**	384	640	896	1920	4352	8064

Table 13 on page 93, Table 14 on page 93, and Table 15 are mostly self-explanatory with the exception of the field of numbers. The core ports and the appropriate value under **Number of ports on edge switch** need to be added in order to determine the total port count. When this is done, a few configurations fall outside of the support envelope of 2048 Nx_Ports. If the number of ISLs is increased between cores and edge switches, then some of the fabrics fall back into the supportable range.

Table 16 on page 95 shows the increase of ISLs between cores and edge switches.

Table 16 Number of ISLs increased between cores and edge switches

Number of ports on core switches	256						
Number of ISLs between core and edge	4	Number of core switches				4	
Number of cores each edge will connect to	2	Number of ISLs between cores				8	
Number of Edge switches	Core ports	Number of ports on edge switch					
		16	24	32	64	140	256
1	**912**	10	18	26	58	134	250
2	**896**	20	36	52	116	268	500
4	**864**	40	72	104	232	536	1000
8	**800**	80	144	208	464	1072	2000
16	**672**	160	288	416	928	2144	4000
24	**544**	240	432	624	1392	3216	6000
32	**416**	320	576	832	1856	4288	8000

Note: Remember, if you are considering deploying a mirrored fabric (and you should because this is a best practice), the number of ports needed on each fabric will be roughly half of the total ports needed.

Once you have decided on a topology, the number and type of switches that will be needed can be determined.

Choosing a switch type

This section provides considerations for choosing a vendor and selecting a model.

Choosing a vendor

Consider the following when choosing a vendor:

- If an environment has standardized on a switch vendor such as Brocade or Cisco, you should use a switch from their product line. Although improvements to test coverage of interop environments have been made, interop fabrics remain the least tested configurations as switch vendors spend much more time verifying interop with their own products rather than investing time in testing interop with another switch vendors' products.

The subject of interoperability is raised because even if the fabrics are not connected when installed, there is a chance that connecting them together will be desired at some point in the future.

- An equally-important reason for using the same vendor is training. A user who has standardized on a particular vendor is less likely to need training on the product. Typically, their expectation of the product's performance is more realistic and any infrastructure challenges (power, monitoring) have already dealt with.
- Sometimes it is not possible to keep with the same vendor as the decision has made to migrate to another vendor. If this is the case, read on, but refer to the "Heterogeneous switch interoperability" section in the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab, for more specific ideas on how to migrate when the need eventually arises.
- If a particular vendor has not been standardized, then determine which features will work best for you.

Selecting a model

Once a vendor has been chosen, it is time to select a model. There are many different aspects to consider, but this section is only in regards to port count.

Switches provide between 8 and 64 ports of connectivity. Directors provide between 8 and 528 ports of connectivity. Keep in mind when ordering a director that they all have minimum shipping configurations. For example, assuming 4 GB/s FC will be used:

- For Cisco: 9513, 9509, and 9506, there is a minimum of one blade per chassis (16 ports).
- For Brocade Silkworm 48000, there is a minimum of two blades per chassis (64 ports).
- For Brocade M Series Intrepid 10000, there is a minimum of two LIMs per chassis (64 ports).
- For Brocade M Series Intrepid 6140, there is a minimum of two UPMs (8 ports).

Factor these minimums in when considering which switch and how many of each to purchase.

3

FC SAN Concepts

The purpose of this chapter is to familiarize the reader with how SAN technology works, rather than how to use the technology. With this as our goal, this information is organized to follow the flow of data from the initiator across the SAN to the target..

Fibre Channel standards

This section provides the following information:

Overview

Fibre Channel standards define layered communications architecture similar to other networking environments. Each level of the Fibre Channel protocol stack provides a specific set of functions, summarized in Figure 30.

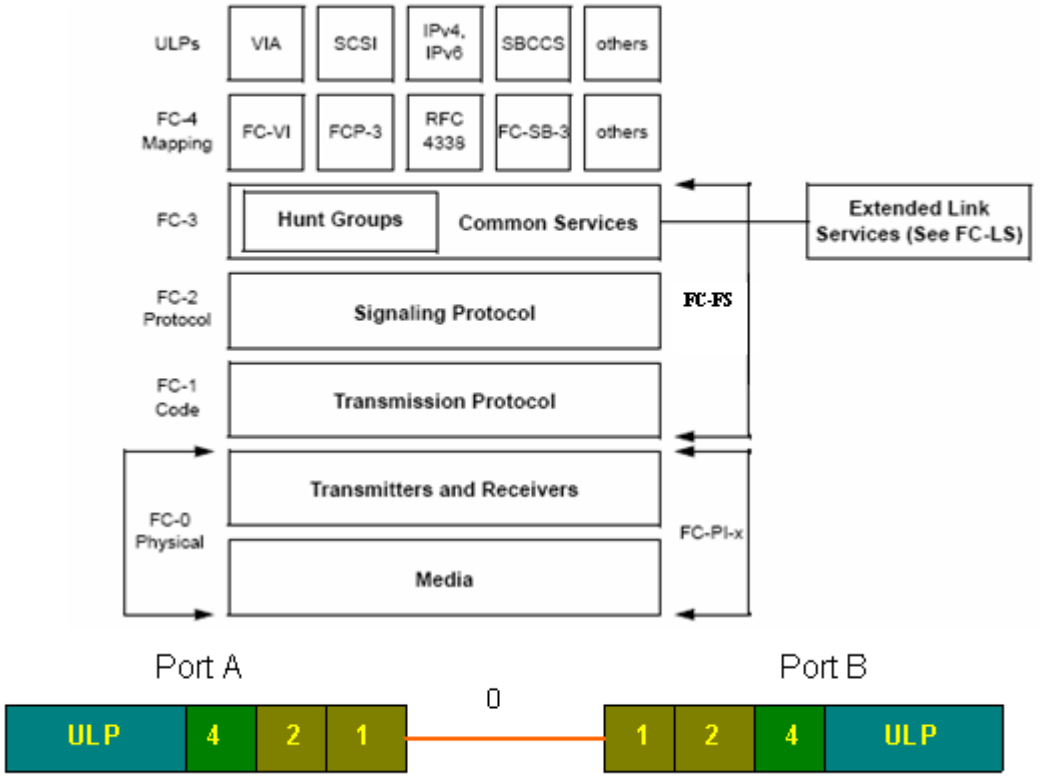


Figure 30 Fibre Channel levels

The highest level in Fibre Channel, FC-4, defines the interface between upper-level protocols and the underlying transport layer. This section primarily focuses on SCSI-3 Fibre Channel Protocol (SCSI-FCP) implementations, which provides a connectivity technology with the channel benefits of low protocol overhead and high performance at the upper level, and the network benefits of extended distance, dynamic configuration, and open systems connectivity at the transport layer.

For more information on Fibre Channel levels, refer to "Architectural layers," next.

Architectural layers

The Fibre Channel architecture covers all aspects of Fibre Channel from the physical interface to the transport of multiple upper layer protocols. What must be understood here is that Fibre Channel is a layered approach and each level has multiple functions each of which describes a specific aspect of Fibre Channel.

The following levels are discussed in this section:

Figure 31 shows the architectural levels in Fibre Channel.

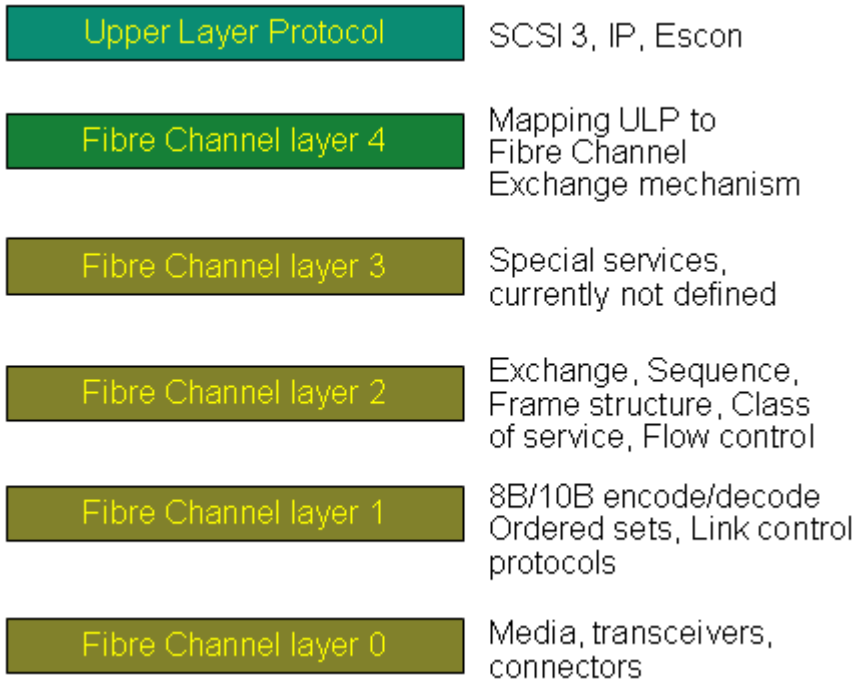


Figure 31 Fibre Channel architectural layers

The standards document Fibre Channel Physical Interface (FC-PI) defines the FC-0 and FC-1 levels of Fibre Channel. FC-FS (Fibre Channel Framing and Signaling) defines the FC-2 level. FC-0 through FC-2 are available on the port while the FC3, FC4, and ULP (Upper Layer Protocol) layers are provided by the system/node being used. Each of these layers will be discussed next in detail to give a clear understanding as to how they all combine to provide the Fibre Channel architecture.

FC-0 layer

The FC-0 layer describes the physical interface including transmission media, transmitters and receivers, and their interfaces. This layer defines data rates provided by the Fibre Channel standard, optical and electrical media that can be used at each rate, connectors associated with each media type, maximum distance capabilities, and other characteristics such as wavelength of light and light levels.

FC-1 layer

The FC-1 layer in Fibre Channel defines how data is encoded prior to transmission and decoded upon receipt. There are three primary functions of the FC-1 layer:

- "Encoding/decoding," next
- "Ordered sets" on page 102
- "Link initialization" on page 104

Encoding/decoding

In order to transmit data over a high-speed serial interface the data is encoded prior to transmission and decoded upon receipt. All information transmitted over Fibre Channel is encoded into 10-bit transmission characters prior to transmission and decoded back into 8-bit bytes at receipt. Encoding the data improves the transmission characteristics of the serial bit stream and facilitates successful recovery of the data at the receiver. This encoding also maintains a balance between the number of ones and zeros transmitted, which allows for the recovery of a clock from the data stream and allows the receiver to be properly biased. Refer to "8b/10b encoding and decoding" on page 115 for more information.

Ordered sets

Fibre Channel uses a number of transmission words to perform control functions. These transmission words consist of four transmission characters, the first of which is the K28.5 special character. The remaining three characters are data characters (D. xx.y) that are used to define the meaning of the ordered set. Three classifications of ordered sets are defined:

- Frame delimiters — Identify the start and end of a frame. There are two types:
  - SOF delimiters — Identify the start of a frame and the class of service associated with the frame. Used to establish a class 1 connection and to signify the beginning and continuation of a sequence.
  - EOF delimiters — Identify the end of a frame. Used to end a connection, to signify whether the frame is the last frame in a sequence, and to indicate certain frame errors.

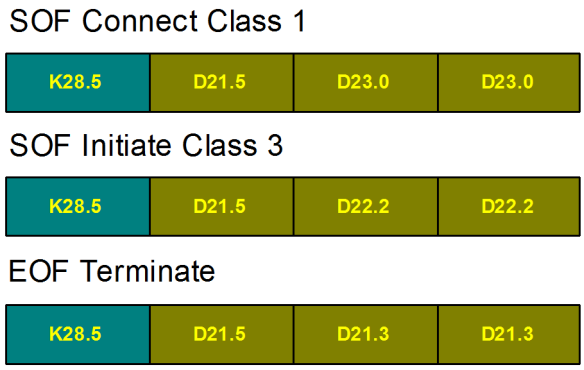


Figure 32 Examples of SOF and EOF delimiters

- Primitive signals are defined for use over a single link for indicating events at a transmitting port. There are three basic types:
  - Arbitrate (ARBx) ordered set is used to indicate that a loop port requires access to the loop.
  - An IDLE indicates the port is ready for frame transmission and reception. They are transmitted when the port has no other specific information to send.
  - The R_RDY ordered set is used to control the transmission of frames on a link and indicates the receiver has emptied a receive buffer and is ready to receive another frame.

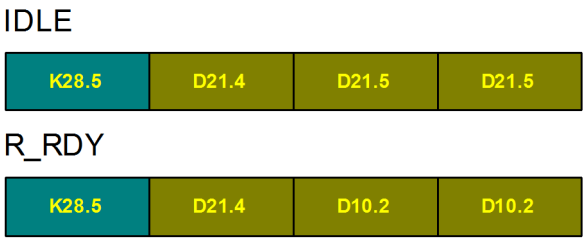


Figure 33 Example of primitive signals

- Primitive sequences ordered sets are used for link initialization and error recovery in Fibre Channel environments. They are used in arbitrated loop and also in switched fabric as for link initialization. Refer to "Link initialization," next.

Link initialization

The four primitive sequences used in link initialization in switched fabric are:

- Non Operational (NOS) — Transmitted by a port to indicate that the transmitting port has detected a link failure.
- Offline (OLS) — Transmitted by a port to indicate the port is beginning link initialization, has received and recognized the NOS sequence, or the port is entering the offline state.
- Link Reset (LR) — Used to indicate a link reset.
- Link Reset response (LRR) — Transmitted by a port to indicate that it has recognized a LR sequence and performed the appropriate actions.

The flow of the primitive sequences in initializing a link is shown in Figure 34 on page 104.

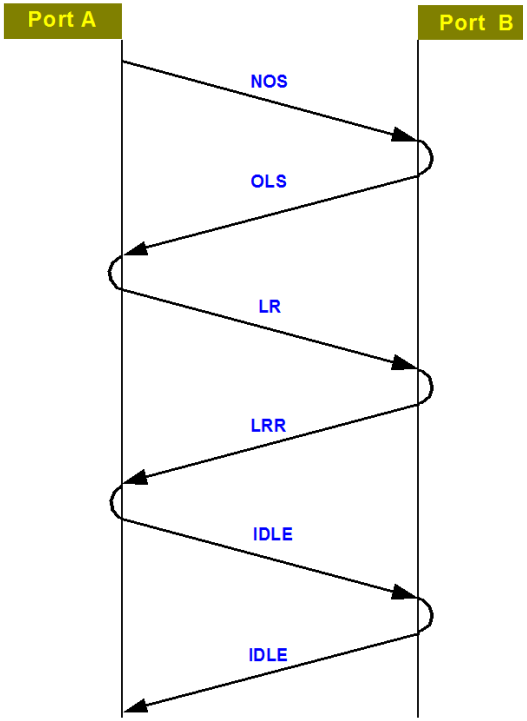


Figure 34 Link initialization handshake

FC-2 layer

The FC-2 layer in Fibre Channel defines the structure and organization of information being delivered and how it is controlled and managed. The FC-2 layer covers exchange and sequence management, frame structure, class of service, and flow control.

There are four tiers to the organizational structure in FC-2 to control and manage delivery of information, as shown in Figure 35.

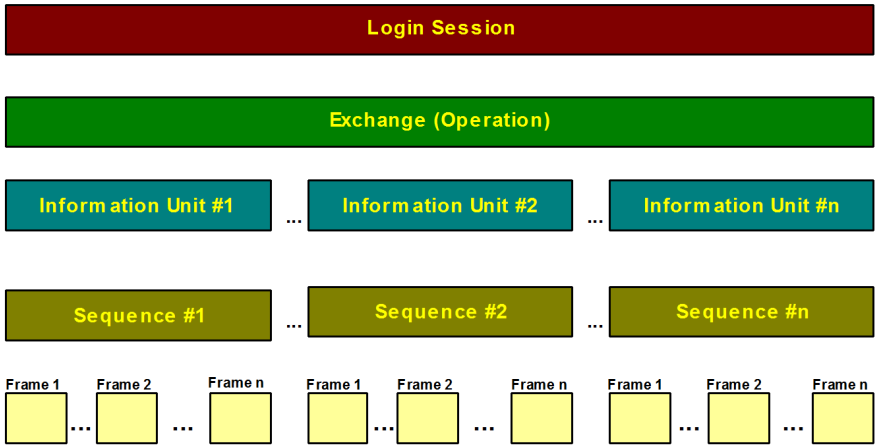


Figure 35 Exchange sequence and frame relationship

Login session tier

Before any I/O type operations can take place in Fibre Channel, two ports must establish a login session. During the login process information is exchanged between both ports, which is used in any further communication between the ports. As long as the login session is active between both ports normal I/O operations may take place. If the login session is lost for any reason then any current I/O is terminated and no subsequent I/O can take place until a new login session is established.

Exchange tier

The exchange is the mechanism that allows two Fibre Channel ports to identify and manage a set of information units. These information units may represent an entire operation (Command, Data, Status) or just part of an operation. Each protocol has its own pieces of information that must be sent to another port in order to perform a certain operation. These protocol specific pieces of information are called *information units*. The structure of these information units is

defined in the FC-4 protocol mapping for the protocol. If an information unit needs to be sent from one port to another as part of an I/O, the request is passed to the layer beneath. Since there are no services defined in the FC-3 layer the information unit is converted into a sequence which is handled at the FC-2 layer. In effect the information unit corresponds to an FC-2 Sequence.

An exchange is composed of one or more sequences (information units) and within the exchange information units can be sent in either of two ways:

- Unidirectional exchange — Information units are sent in one direction only from exchange originator to exchange responder.
- Bidirectional exchange — Information units are sent in both directions during the course of the exchange.

When the exchange is created the originator has the initiative to send the first sequence but once that sequence has completed the originator may transfer control (initiative) to the responder so the responder can send a sequence in return. This is known as *transferring sequence initiative*.

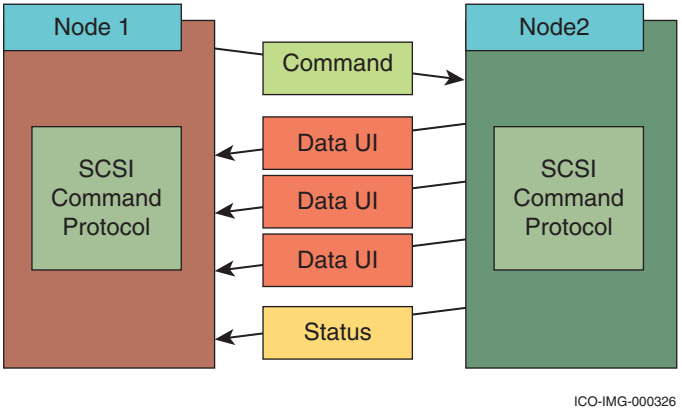


Figure 36 The Fibre Channel exchange

Exchange Identification — Since a port may have multiple exchanges open at the same time there must be a way to identify each exchange individually. Two exchange identifiers exist to provide this:

- Originator Exchange Identification (OX_ID)

When a port wants to begin an I/O with another port it has to originate an Exchange. The exchange originator assigns an OX_ID to this exchange that has to be unique to the originator-responder pair. Once the exchange is created and the OX_ID is assigned the originator can begin transferring the first sequence to the responder. To identify frames in the exchange this OX_ID is included in every frame in the exchange whether it is sent by the originator or the responder.

One factor to consider is that if multiple originators communicate with a responder as would be the case in a SAN environment with multiple HBAs logging in to a Symmetrix FA it is possible that multiple originators may assign the same OX_ID to separate exchanges. In this case the responder uses the OX_ID in conjunction with the source address identifier (S_ID) of the originator to uniquely identify the exchange.

- Responder Exchange Identification (RX_ID)

  When the responder receives the first frame of an exchange, in order to manage the operation it assigns its own identifier to the exchange which is the RX_ID. This is not required by the standard and if the field in the frame header contains 'FFFF' it is not being implemented. If it is implemented the responder will include the RX_ID value in every frame sent for that exchange.

Sequence

When information needs to be sent from one port to another a request is made to the FC-2 layer to deliver this information. This is accomplished by putting this information into a sequence of frames (refer to "Frame" on page 109). The maximum amount of data that can be transmitted by a single frame is 2112 bytes, thus some sequences will contain more than one frame.

There are number of terms that need to be understood when dealing with sequences:

Sequence Initiator	The sequence initiator is the N_Port that initiates a sequence and transmits the data frames to the destination port.
Sequence Recipient	The sequence recipient is the N_Port that is receiving the Data frames from the sequence initiator.

Sequence ID (SEQ_ID)	Because a port can have multiple open exchanges there then exists the possibility of having multiple open sequences. Each sequence needs its own identifier that is the SEQ_ID, which is a value set in the frame header of every frame in that sequence.
Sequence Count (SEQ_CNT)	Each frame in a sequence is sequentially numbered by the sequence initiator. This sequential number is in the frame header in the SEQ_CNT field. The SEQ_CNT indicates the position of the frame in the sequence and is also used in verifying every frame of a sequence has been received and is used in handling responses such as acknowledgements, rejects, or busy being returned for a certain frame.

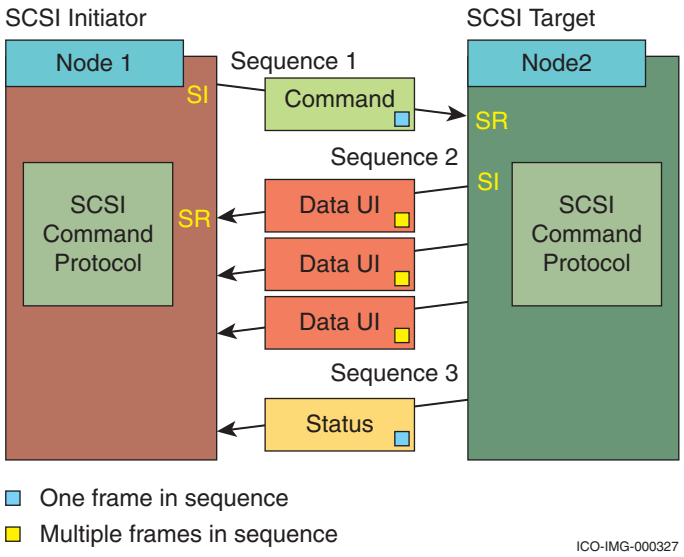


Figure 37 Sequences in an Exchange

Figure 37 shows multiple sequences in an exchange showing sequence initiative (SI) and sequence recipient (SR) changing between Initiator and Target during the exchange. Frames in the same sequence will have the same SEQ_ID. Frames in the same sequence will have their own unique SEQ_CNT value.

Frame

Information transferred in Fibre Channel is packaged into frames. Frames are structured according to a defined format and consist of a fixed length header, a variable length data field (also known as PAYLOAD) and a fixed length CRC. The beginning of a frame consists of a Start of Frame delimiter and at the end there is an End of Frame delimiter (refer to page 102). The header contains address and control information. For more information on frame structure, refer to "Frame structure in Fibre Channel" on page 272.

FC-3 layer

The FC-3 layer provides for future functions that may be implemented in Fibre Channel. These functions are known as *Common Services*. The FC-3 layer could also be used to provide data compression and encryption prior to delivery to the FC-2 layer.

FC-4 layer

The FC-4 layer defines the mapping between the protocols that can be transported by Fibre Channel and the lower transport layers of Fibre Channel (FC0, FC1 and FC2). Each Upper Layer Protocol has its own specific command, data, status or packet information that needs to be communicated with other nodes in order for the protocol to operate. The FC-4 layer defines the format and structure of the protocol specific information being delivered by Fibre Channel.

Examples of protocols that have been mapped to Fibre Channel are:

- Fibre Channel Protocol for SCSI-3 (SCSI-FCP)

  The SCSI FCP protocol mapping defines how SCSI-3 protocol operates using the Fibre Channel interface. It includes for example how SCSI CDBs, Logical Unit Number (LUN), SCSI data, SCSI status, and sense information are structured and transported over the Fibre Channel interface.

- Fibre Channel Single-Byte Command Code Set-2 (FC-SB-2)

  This is a new mapping of the ESCON protocol for transport through Fibre Channel.

- Fibre Channel Link Encapsulation (FC-LE)

  This mapping can be used for different network protocols although it is mainly focussed on transport of TCP/IP.

Upper Layer Protocol

The Upper Layer Protocol (ULP) is not really a part of the Fibre Channel protocol as such, but is the command set or packet architecture that will be transported by Fibre Channel.

Examples of Upper Layer Protocols are:

- SCSI
- ESCON
- Intelligent Peripheral Interface (IPI)
- High Performance Parallel Interface (HiPPI)
- IEEE 802. 2 Logical Link Control (LLC), which defines logical content of information transported over networks

Hosts

In the context of an FC SAN, a host services requests from applications. As a part of servicing these requests, a host may need to store or retrieve data to or from a storage array, located somewhere on the SAN. There are different kinds of hosts, but they can be broken down into two main categories, *mainframe* and *open systems*.

- Mainframe

  A mainframe host connects to the SAN via Channels, accesses storage behind Control Units, and usually uses the SBCCS (Single Byte Command Control Set) protocol.

- Open systems

  An open system host, also known as a server, connects to the SAN through HBAs (Host Bus Adapters), accesses storage behind targets, and usually uses the SCSI-FCP protocol.

This section will mainly focus on open system hosts. The following host design consideration is discussed in this section:

- "Fan-in and fan-out considerations" on page 111

Fan-in and fan-out considerations

In the Symmetrix environment, *fan-in* and *fan-out* mean *into* and *out of* the Symmetrix system. E-Lab Navigator contains recommended fan-in and fan-out ratios. Figure 38 shows a fan-in rate of 1:4.

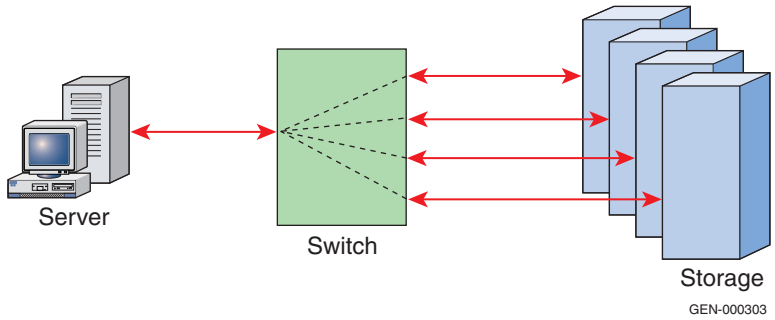


Figure 38 Example of fan-in

Figure 39 shows a fan-out rate of 4:1.

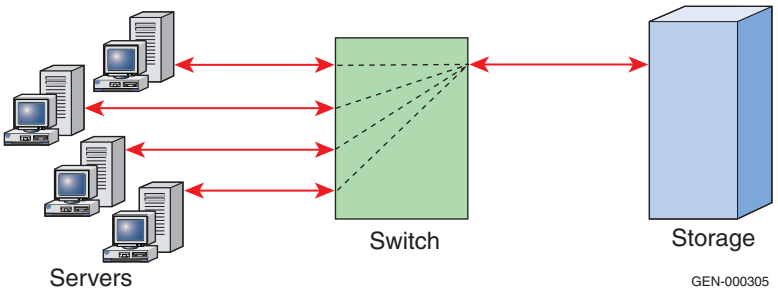


Figure 39 Example of fan-out

HBAs

In the context of an FC SAN, HBAs (host bus adapters) are used to efficiently connect the host to the SAN. Procedures for configuring most EMC-qualified HBAs for connection to EMC storage arrays can be found at the HBA vendors' websites.

Emulex

To download the Emulex documentation:

1. Access http://www.emulex.com.
2. Click **drivers, software, and manuals** at the left side of the screen.
3. Click **EMC** the upper center of the next screen.
4. Click your HBA on the list at the left side of the screen.
5. Under **Drivers for (your OS)**, click **Installation and Configuration** in the **Online Manuals** column.

QLogic

To download the QLogic documentation:

1. Access http://www.qlogic.com.
2. Click **Downloads** at the left of the screen.
3. Click **EMC** to the right of **OEM approved/recommended drivers and firmware**.
4. Find the description of your HBA driver in the **Name** column of the table for your HBA model. Then click the **Readme** link in the associated **Description** column.

Brocade

To download the EMC-supported drivers and documentation:

1. Access www.brocade.com/adapters.
2. Select the adapter type you are looking for.
3. Click **Downloads**.
4. Under **OEM Models**, select the EMC link.

Application Specific Integrated Circuits (ASICs)

An ASIC is a highly-specialized integrated circuit designed to perform a single task extremely well. Although ASICs can be used for many different purposes in FC HBAs, switches, or storage arrays, the ASICs referred to most frequently are the Port ASICs. A Port ASIC is an ASIC that at a minimum controls basic port behavior and is arguably the largest factor in determining an FC Port's basic set of behaviors or *personality*. It is commonly accepted that switches using the same Port ASIC will react almost identically under the same set of conditions. Because of this, switches are usually classified by the Port ASIC used in its design. Switches that utilize the same Port ASICs are said to be members of the same ASIC family. Some current examples of Port ASICs are Brocade's Condor and Cisco's Vegas ASICs.

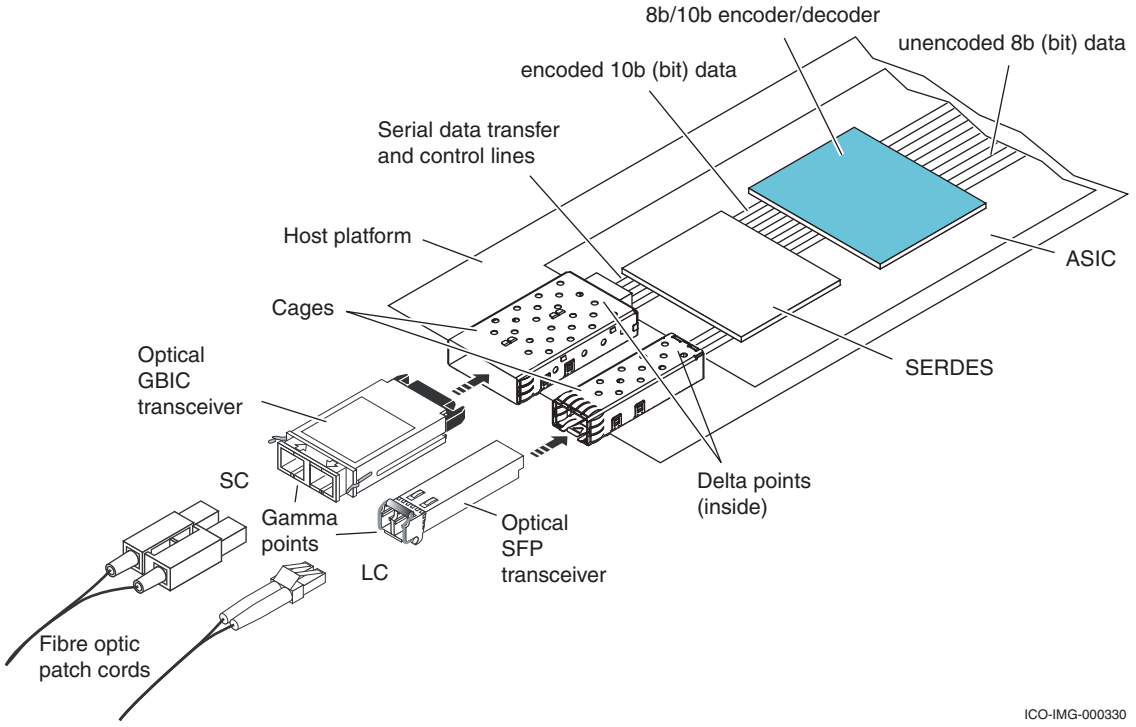


Figure 40 ASIC

8b/10b encoding and decoding

In the context of an FC SAN, 8b/10b encoding is the process of converting 8 bits of data into a 10-bit representation (Figure 41).

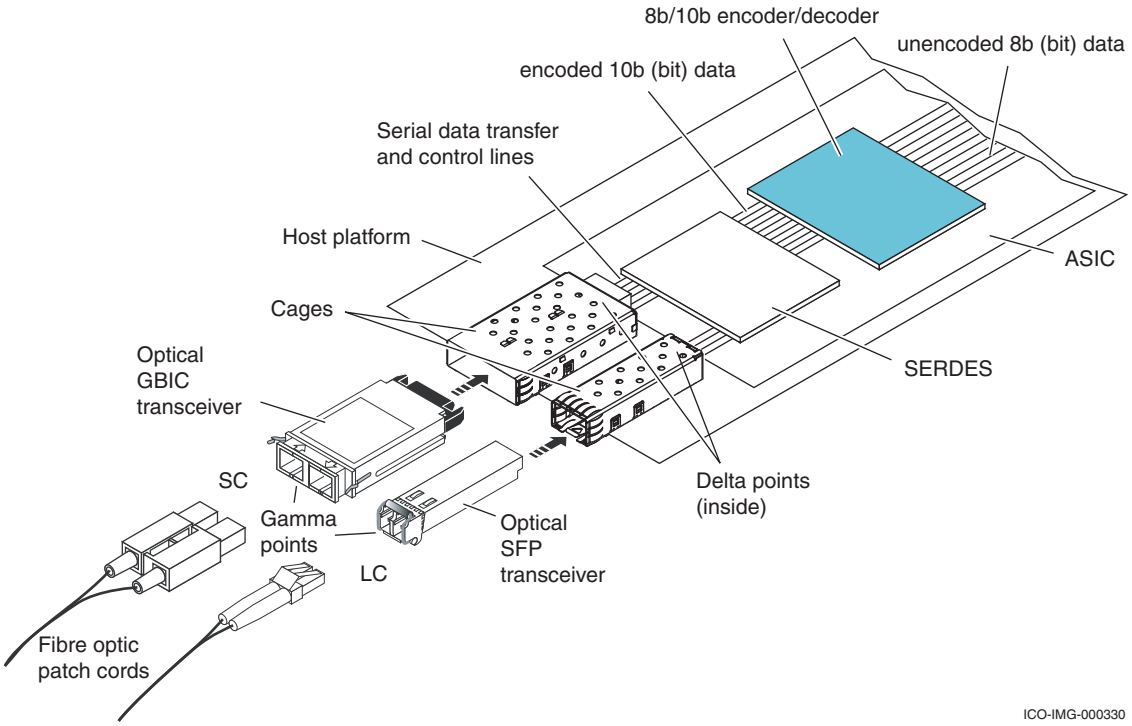


Figure 41 8b/10b Encoder/decoder

There are several reasons for this conversion and one of the most important is that it enables the receiver to recover a clock signal from the data stream. Other reasons include data integrity and ensuring that there are no excessive DC components, a problem specific to implementations that use copper wire instead of instead of Optical Fiber. The 8b/10b encoding and decoding logic are usually built into the port ASIC.

A 10-bit code facilitates up to 1024 unique bit patterns. From these patterns 256 data characters are encoded in one or two 10-bit patterns. (For more information, refer to "Disparity" on page 118.) A number of data patterns are allocated to special characters, which will be discuss shortly.

To explain this, the following is an example of a byte (0xF4) that we would like to send to the other side:

0xF4 = 1111 0100

The bit order gets reversed and the bits are divided into a 5-bit and a 3-bit block:

abcde fgh

00101 111

This pattern is now represents the data character D20. 7, which must be encoded depending on the current running disparity. Figure 42 shows how this example arrives at the D20. 7 data character.

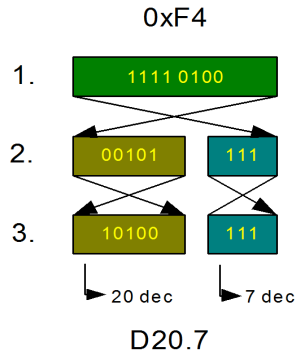


Figure 42 Example

1. Shows the binary bit pattern of our 0xF4.
2. First the bit pattern is reversed and then divided into the 5- and 3-bit sub-blocks.
3. Shows how this example arrives at the data character 20.7.

Note: In reality the "D" (data character) is achieved by a high or low control line input to the encoding circuitry. The same method would be used for the 'K' special characters.

Figure 43 shows an example of a number of data characters being transmitted. The data byte of 0xF4 is transmitted as an encoded byte.

How this bit pattern was arrived at is discussed in "Disparity" on page 118.

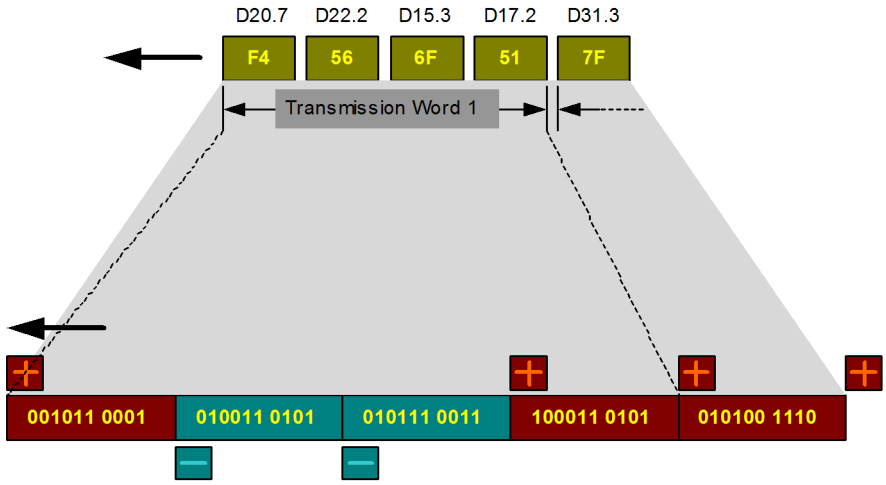


Figure 43 Transmitting 10 bit encoded bytes

All bytes are transmitted as transmission characters, four of which form a transmission word. These transmission words are either *ordered sets* or data *transmission words*. A transmission word is the smallest unit of transfer in Fibre Channel. The '+' and '-' above are an indicator of the balance of ones / zeros within the most recent encoded character. "Ordered sets" is discussed on page 102. If one had to transmit a SCSI block of zeros, it should be obvious from Figure 43 why the encoding is needed. In a serial transmission stream the receiver would have a hard time detecting the individual bytes and knowing for sure whether it received 511 0x00s or 512.

Figure 44 shows what the difference is between an ordered set and a data transmission word.

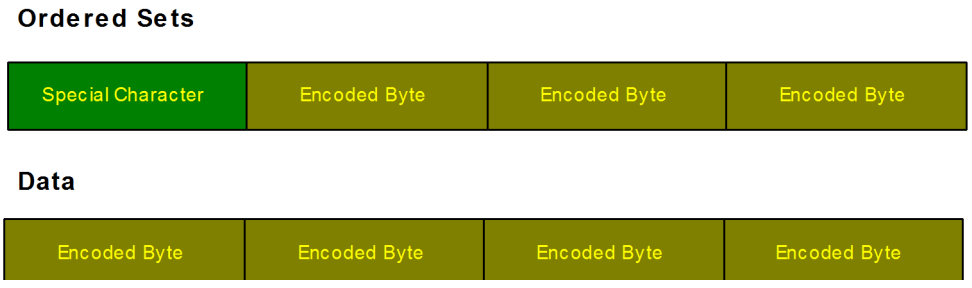


Figure 44 Transmission words

Disparity — The encoding process converts 8-bit input characters to 10-bit transmission characters. To prevent excessive DC components in the bit stream (ensure the same average amount of 1 bit and 0 bits) only certain characters are used. Only characters containing six ones and four zeros, five ones and five zeros, or six zeros and four ones are allowed.

Disparity is the ratio of 1 bits to 0 bits in the bit stream. The possibilities are:

- Positive disparity — Transmission character has more 1 bits than 0 bits
- Negative disparity — Transmission character has more 0 bits than 1 bits
- Neutral disparity — Transmission character has equal number of 0 and 1 bits

Note that each character that has an unequal number of 1 bits and zero bits has two different encoding patterns. One pattern contains six ones and four zeros and the other contains six zeros and four ones.

When the encoder creates a character with an unequal number of zeros than ones, it keeps track of a variable called *Current Running Disparity*. This variable is set to *positive* if the character generated has more ones than zeros and is set to *negative* if the character generated has more zeros than ones. This variable is used to select the appropriate encoding of the next character (positive or negative) to balance the number of 1 bits and 0 bits being transmitted.

For example, the hex value 0xF4 and its two encoding values depending on the value of the Current Running Disparity are shown in Figure 45.

		Current RD -	Current RD+
Data Byte Name	Bits HGF EDCBA	abcdei fghj	abcdei fghj
D20.7	111 10100	001011 0111	001011 0001

Figure 45 Data character “F4”

This code would be automatically generated by the encoding hardware taking the following input variables:

- 5- and 3-bit sub-blocks
- D or K indicator

- Current Running Disparity

Figure 46 shows the transmission of encoded data and the Current Running Disparity changing depending on the transmission character being sent. The positive and negative transmission characters are also shown in this diagram.

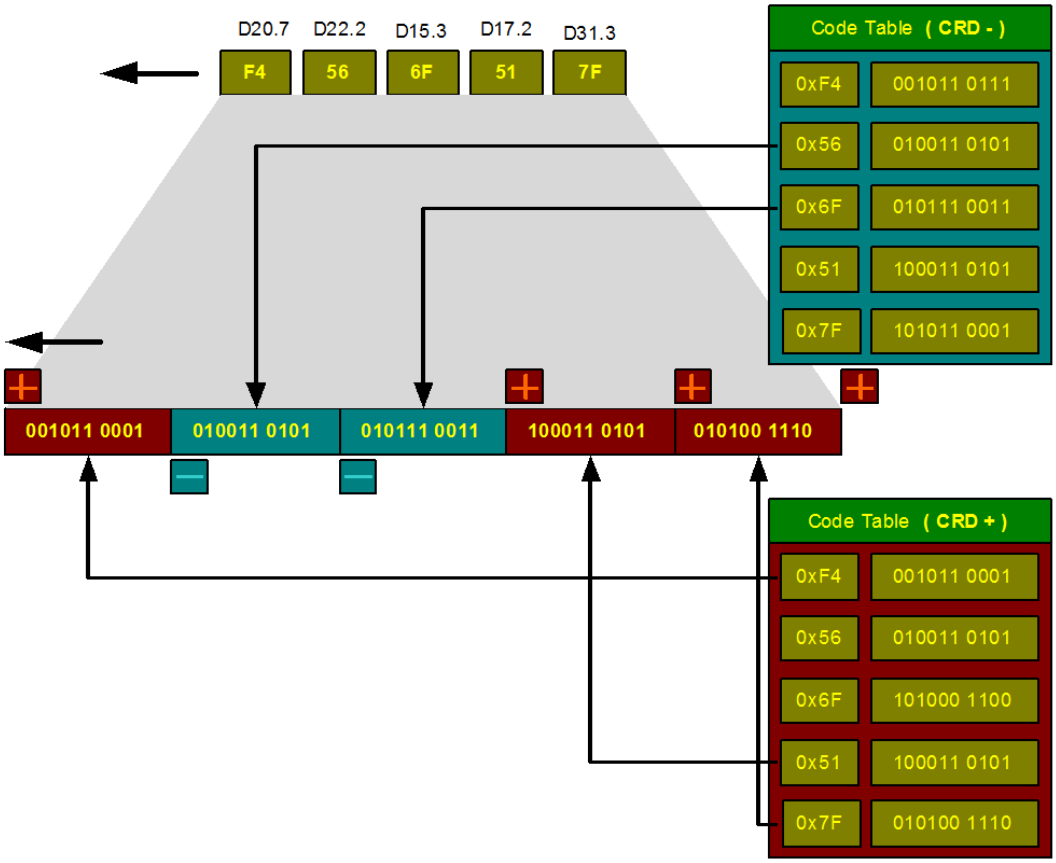


Figure 46 Sending encoded data

Special characters are derived in a similar fashion but always start with a "K". There are 12 special character defined in the 8b/10b encoding scheme but only the **K28. 5** special character is used by Fibre Channel. Its use is limited to the first character in a transmission word known as an Ordered Set (see "Ordered sets" on page 102).

64b/66b encoding

64b/66b is an encoding scheme similar to 8b/10b and is used with 10 G FC, 10 GbE and most recently with 16 G FC. The main difference with 64b/66b is that only 2 bits out of every 66 are used for overhead, whereas with 8b/10b, 2 bits out of every 10 are used for overhead.

By using 64b/66b, the usable bandwidth of the link is 97% of the actual link speed, while with 8b/10b, the usable bandwidth is 80% of the link speed.

By using 64b/66b encoding, the manufacturers of 16 G FC hardware were able to decrease the signaling rate necessary to carry 1600 MB/s of data. This decrease in signaling rate made it possible to release 16 G FC earlier. It also made the components less expensive.

64b/66b uses special control characters (primitives); but an important distinction is that with 64b/66b, the primitives necessary to support FC-AL were never defined. As a result, 10 G FC and 16 G FC do *not* support FC-AL.

SERDES

The purpose of the SERDES is to convert 10-bit parallel data into 10-bit serial data. The SERDES is usually built into the port ASIC (Figure 47).

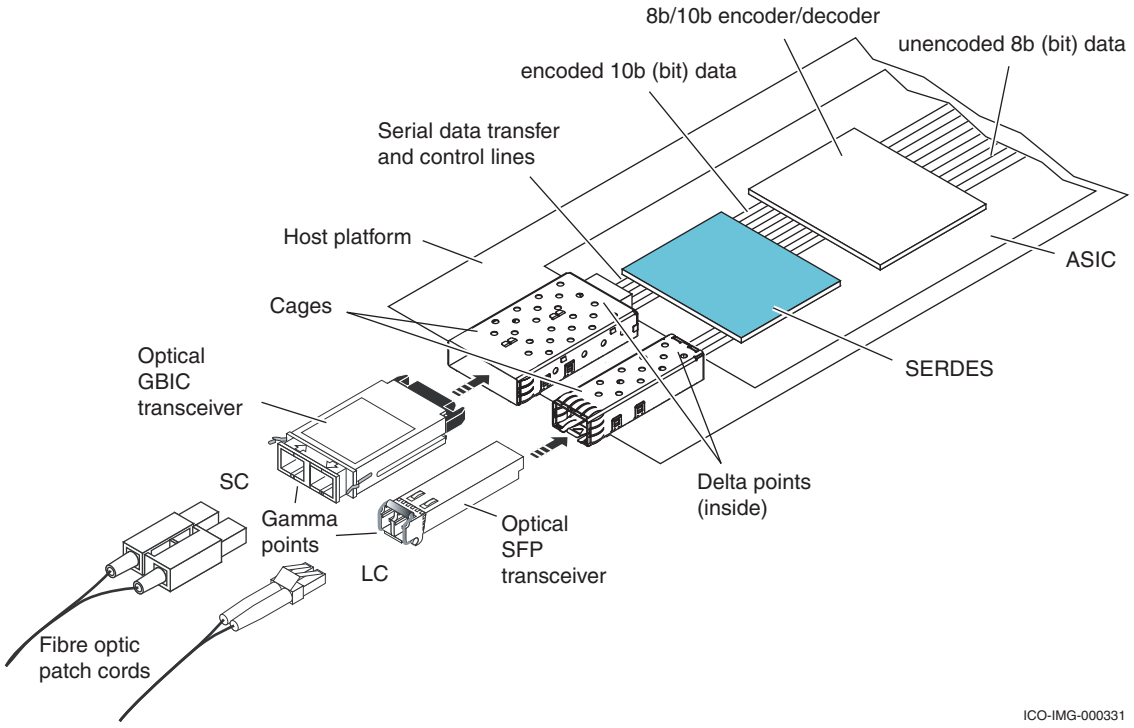


Figure 47 SERDES example

Optics

Optics (formerly known as optical transceivers) convert the high and low electrical pulses coming from the SERDES (serializer / deserializer) on the host platform, into high and low light levels that propagate down the optical medium (fiber). Optics also convert incoming high and low light intensity levels into high and low voltage outputs, which will be interpreted by the SERDES on the receiver before being passed to the 8b/10b decoder on the receiver, as shown in Figure 48.

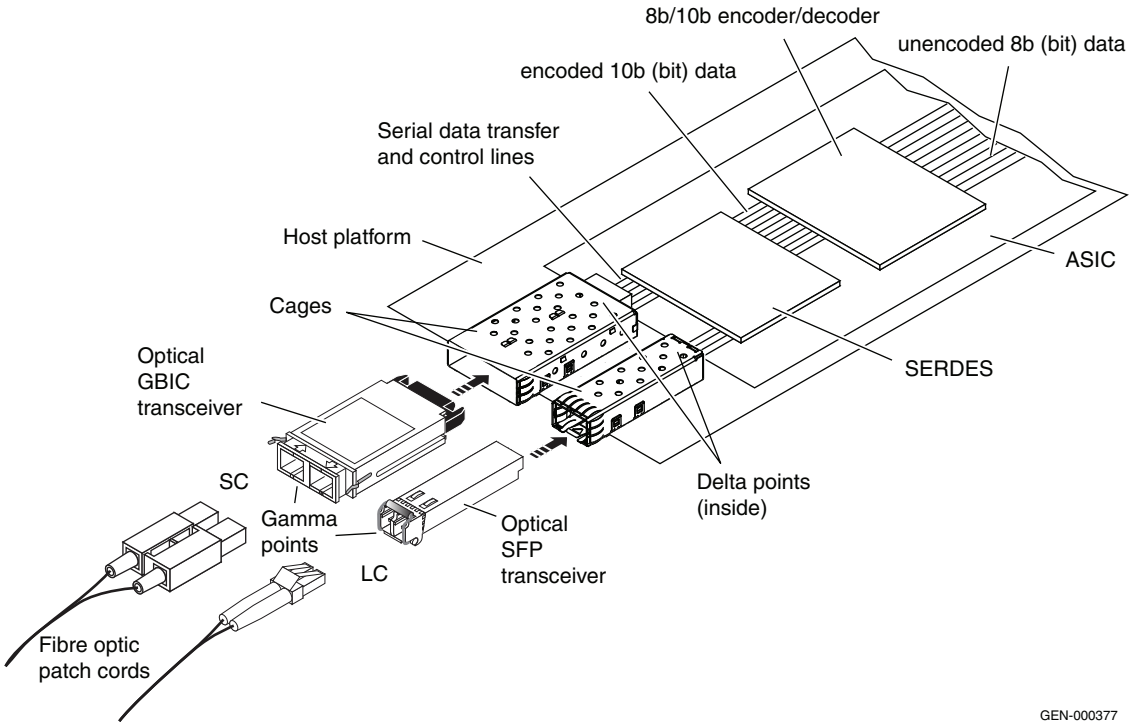


Figure 48 Optics

Optics come in a variety of packages (SC, LC, XFP, etc). Each package typically has its own cable type. Packages of the same type are usually interchangeable because they are built using the same set of standards that define the connector layout and other necessary information. SFPs, for example, are expected to adhere to the SFP multisource agreement.

Some important points about optics in general are:

- Optics are usually interchangeable with each other, but not always.

  Even though optics are built using the same standards, issues may be encountered due to the physical characteristics of the optic that the developer of the platform, into which the optic will be inserted, may not have considered. These problems may manifest as an inability to bring a link up (gain bit sync) or as periodic CRC errors.

  Note: Because of the potential for interoperability issues in this area, it is recommended that should it be necessary to replace a failed optic, you replace it with one of the same type, or one that has been tested and is approved for the host platform.

- Optics convert an electrical signal to an optical signal and conversely.

  The electrical interface of the optical transceiver has many different connection points (pads), (labeled as delta points in Figure 48 on page 122), which are used to connect to the host platform (i.e., FC switch, HBA, front end storage port). The host platform will generate a signal (a high voltage or low voltage) at some of these connection points to cause the optic to do something (i.e., make light brighter, turn off light). The optic can also provide a high or low signal at some of these connections to indicate to the host platform that an external condition, such as loss of light, has been detected.

  The data transferred between the SERDES and the optic is in a serial 10-bit format and it not interpreted by the optics in any way. This means that bit and CRC errors are not detected by the optics, and in fact are not detected until the after SERDES when the 10-bit data is being decoded.

  Note: Although optics are not capable of detecting CRC or bit errors, they are frequently the source of them.

- Some SFPs provide extended digital data.

  Most of the optics shipping today provide extended digital diagnostics information which can be accessed if the host platform supports reading this data and provides an interface to display it. Some of the information that is provided by the SFP, and what it may indicate, is provided below.

  - Temperature: The operating temperature of the optic
  - 3.3 Voltage: The internal voltage of the optic
  - Current: The current being drawn by the optic

  These measurements are specific to the optic and informational in nature. If you were to monitor each optic and were to observe a steady change in one of these values, it may be possible to predict an optic failure before it were to occur. By comparing all of the values for the same type of optic installed in the same host platform, a single optic that is operating with a set of different values could be detected and may be in the process of failing.

  Be aware, though, that these changes might not mean anything; for example, the values might be fluctuating due to changing environmental conditions. Do not worry about these values unless they are approaching a warning or alarm level. Once an optic reaches one of these thresholds, you should consider replacing it.

  The next values are provided in microwatts (mW) or dBm, depending on the platform, and can be extremely useful when debugging why a link will not come up.

  - TX Power: A measurement of the strength of the signal being transmitted by the optic.
  - RX Power: A measurement of the strength of the signal being received by the optic.

  In order to interpret the TX and RX power values, it will be useful to quickly review microwatts and dBm and determine how they are related.

  - Microwatts (mW): As used in this example, mW refers to a measurement of light intensity.
  - dBm: dBM is an abbreviation for the power ratio in decibels (dB) of the measured power referenced to one milliwatt.

Converting between microwatt and dBm is fairly straightforward if you have a calculator that supports the log function:

- Bm to mW:
  mW = 10(dBm/10)
- mW to dBm:
  Note the optics measurements are given in microwatts before you can use the following formula, convert the value in microwatts to milliwatts by multiplying by 10-3.

  For example 1000 microwatts becomes 1 milliwatt.

  dBM = 10 * LOG mW

  Instead of converting to get an exact value, use Figure 49 to get an idea of where the measured value falls.

microwatt	milliwatt	dBm	Description
1	0.001	-30.00	Loss of Signal
10	0.01	-20.00	
25.1	0.0251	-16.00	Minimum average RX value allowed at 2 Gb/s
61	0.061	-12.10	Minimum average RX value allowed at 4 Gb/s
76	0.076	-11.20	Minimum average RX value allowed at 8 Gb/s
89	0.089	-10.50	Minimum average RX value allowed at 16 Gb/s
100	0.1	-10.00	Minimum average TX value allowed at 2 Gb/s
125.9	0.1259	-9.00	Minimum average TX value allowed at 4 Gb/s
151.3	0.1513	-8.20	Minimum average TX value allowed at 8 Gb/s
165.9	0.1659	-7.80	Minimum average TX value allowed at 16 Gb/s
200	0.2	-6.99	Typical operating range
250	0.25	-6.02	
300	0.3	-5.23	
350	0.35	-4.56	
400	0.4	-3.98	
450	0.45	-3.47	
500	0.5	-3.01	
550	0.55	-2.60	
600	0.6	-2.22	

Figure 49 **Measured values**

Note: This information can be found in FC-PI-5 table 11, located at http://www.t11.org. Minimum TX values can be found under "Average launched power, min." Minimum RX values can be found under "Unstressed receiver sensitivity, OMA."

The value of -30 dBm shown in the **Loss of Signal** row is the approximate value that a multimode optic will detect a loss of signal. It was determined by reviewing a number of SFP data sheets and should not, therefore, be treated as a universal value.

On all SFPs, pin 8 (LOS) is LOW when a signal is detected and HIGH when a signal is not detected. The point at which this will change state from a LOW to a HIGH depends upon the optical transceiver, but will be around -30 dBm or 0.001 mW. However, well before the link could get all the way down to -30 dBm, the bit error rate would become so high that the link would lose sync.

Please note the following misconceptions:

- A high electrical signal is converted into a bright light pulse and a low electrical signal is converted into dark (transmitter off).
- 8b/10b encoding was created so that a bunch of zeros would not be confused with a loss of signal.

Note: Refer to"8b/10b encoding and decoding" on page 115 for more information.

Both of these are misconceptions. High and low light levels are high and low with respect to themselves and *not* static values, as is the case when working with a digital circuit (high = 5V, Low = 0V).

Note: All of the light level measurements are provided by the optic to the host platform as a value between 0 and 65535, with 0 being equal to -40 dBm and 65535 being to 8.2 dBm.

The values of -9 dBm (shown in the **Minimum TX value allowed at 4 Gb/s row**) and -10 dBm (shown in the **Minimum TX value allowed at 2 Gb/s** row) were taken from FC-PI-2. Both represent the minimum transmit signal allow.

The values of -15 dBm shown in the **Minimum RX value allowed at 4 Gb/s** row) and -16 dBm (shown in the **Minimum RX value allowed at 2 Gb/s** row) were calculated by subtracting the maximum link power budget of 6 dB from the minimum TX value allowed. Both represent the minimum receive signal that needs to be decoded without exceeding the BER of 10^{-12}.

Three examples of host platforms (FC switches in this case) that provide a user interface to display enhanced digital diagnostics information are provided next.

Brocade M-EOS (McDATA) platforms:

Using the command **show port opticdata [port number]**

```
Root> show port opticdata 1
Port Number:       1
Overall Health:    Normal
Transciever:       SFP
Type         Value     Low Warning  High Warning  Low Alarm  High Alarm
-----------  --------  -----------  ------------  ---------  ----------
Temperature  36.769    -20.000      90.000        -25.000    95.000
3.3 Voltage  3.265     2.900        3.700         2.700      3.900
Current      8.374     2.000        14.000        1.000      17.000
TX Power     377.200   79.000       631.000       67.000     631.000
RX Power     294.100   15.800       794.000       10.000     1259.000
```

Brocade FOS platforms:

Using the command **sfpshow [portnumber]**

```
l82c7207:admin> sfpshow 2/20
Identifier:  3    SFP
Connector:   7    LC
Transceiver: 150c402000000000 100,200,400_MB/s M5,M6
   sw Inter_dist
Encoding:    1    8B10B
Baud Rate:   43   (units 100 megabaud)
Length 9u:   0    (units km)
Length 9u:   0    (units 100 meters)
Length 50u:  15   (units 10 meters)
Length 62.5u:7    (units 10 meters)
Length Cu:   0    (units 1 meter)
Vendor Name: AGILENT
Vendor OUI:  00:30:d3
Vendor PN:   AFBR-57R5AEZ
Vendor Rev:
Wavelength:  850  (units nm)
Options:     001a Loss_of_Sig,Tx_Fault,Tx_Disable
BR Max:      0
BR Min:      0
Serial No:   A206080273
Date Code:   060223
Temperature: 38 Centigrade
Current:     5.350 mAmps
Voltage:     3269.2 mVolts
RX Power:    0.0 uWatts
TX Power:    335.5 uWatts
l82c7207:admin>
```

Cisco SAN-OS platforms:

Using the command **show interface fc [slot #/port #] transceiver details**

```
switch# show interface fc 2/7 transceiver details
fc2/7 sfp is present
    name is CISCO-FINISAR
    part number is FTLF8524P2BNL-C2
    revision is 0000
    serial number is FNS0942B0JA
    fc-transmitter type is short wave laser w/o OFC (SN)
    fc-transmitter supports intermediate distance link length
    media type is multi-mode, 62.5m (M6)
    Supported speed is 400 MBytes/sec
    Nominal bit rate is 4300 MBits/sec
    Link length supported for 50/125mm fiber is 150 m(s)
    Link length supported for 62.5/125mm fiber is 70 m(s)
    cisco extended id is unknown (0x0)

    Failed to get runtime SFP state info.
    SFP Detail Diagnostics Information
  ------------------------------------------------------------------------
                                    Alarms                  Warnings
                                High        Low         High          Low
  ------------------------------------------------------------------------
  Temperature  32.92 C          0.00 C      0.00 C      0.00 C        0.00 C
  Voltage       3.30 V          0.00 V      0.00 V      0.00 V        0.00 V
  Current       8.05 mA         0.00 mA     0.00 mA     0.00 mA       0.00 mA
  Tx Power     -4.21 dBm        N/A         N/A         N/A
  Rx Power     -7.08 dBm        N/A         N/A         N/A

  Transmit Fault Count = 0
  ------------------------------------------------------------------------
  Note: ++  high-alarm; +  high-warning; --  low-alarm; -  low-warning
```

Applicable standards /organizations

- FC-PI-2
- TIA FO workgroup
- SFP multi-source agreement

Fiber

This section contains the following information:

Overview

The ports connect to the fibre topology through a link. The link is a cable, or other connection, that carries the data. The ports transmit and receive information through two separate fibers. The link may consist of optical fibers or electrical cable. The transmitters may be either longwave laser, shortwave laser, LED, or electrical.

The fiber construction, as shown in Figure 50, consists of a central core through which the light travels. The core is surrounded by cladding, the function of which is to reflect and contain the light within the core. The core and cladding are made from a glass material and can be easily damaged. To protect the fiber from physical and environmental damage to the core and cladding, it is covered in protective layers which also give the fiber considerable strength.

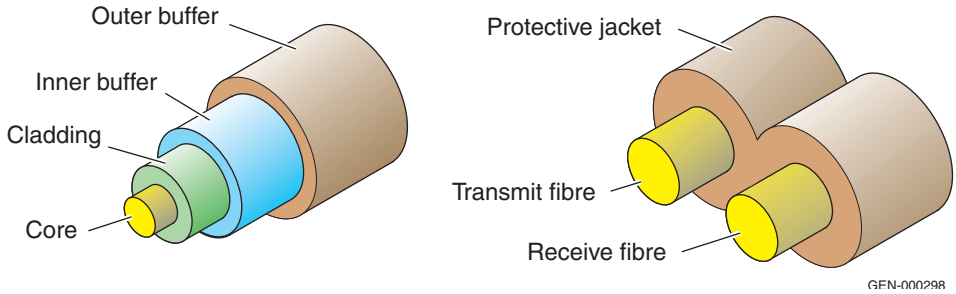


Figure 50 Fibre construction

There is a minimum bend radius specified around which the fiber may be bent that, if exceeded, may result in a degradation in transmission over the fiber or, worst case, may result in possible damage to the fiber itself. In normal use these cables are quite robust and do not require any special handling apart from observing the minimum bend radius requirement.

The core and cladding diameter (in µm) is typically the means by which a fiber is specified. For example, 62.5/125µm fiber has a core diameter of 62.5µm and a cladding diameter of 125µm. Two of these fibers are normally combined in one *duplex* cable that is terminated at both ends with a duplex connector. The two fibers are used to send data in opposite directions for transmit and receive. Simultaneous transmission/reception is possible over the duplex cable.

Single mode

There are two modes of transmission in Fibre Channel: single-mode and multimode.

Single mode links have a fiber core of 9-10 um and use a long wavelength laser operating in the infrared portion of the spectrum at 1300 nanometers (nm) as the light source. This light is not visible to the human eye. The lower core diameter enables single-mode links to support a maximum distance of 10 km between ports as all the light propagates along the same path in the fiber as is shown in Figure 51 on page 131. Single-mode links are mainly used over long-distance transmissions and are implemented on certain versions of the Symmetrix Fibre Channel adapter. For example, the 201-333-901 is a four-port 2 Gb/s capable adapter that has three multimode and one single-mode port

Multimode

Multimode is less costly than single-mode and is used where the distance capabilities of single-mode are not required. The Fibre Channel links are usually based on either a 50 or a 62.5 um core diameter and support light wavelengths of approximately 800nm. This increased core diameter size, as opposed to single-mode, means the light has multiple propagation modes (paths) through the fiber. Therefore, some of the light of a pulse may take one path through the fiber and the rest of the light take another path. The result is called *Modal Dispersion*. This results in the spreading of the pulse which limits the distance that can be achieved with multimode cabling.

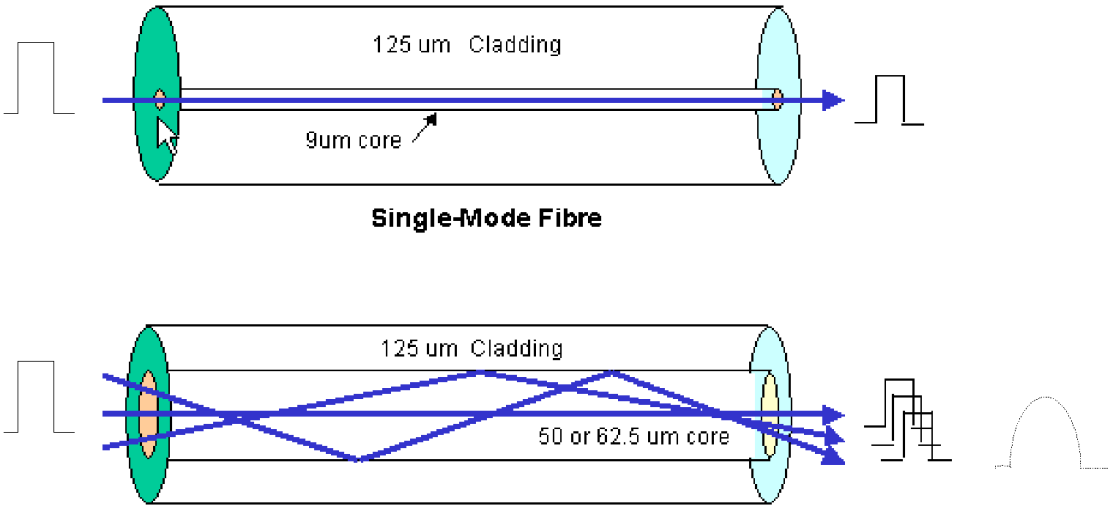


Figure 51 Single and multimode optical fibre

Link loss budget

Historically, link loss budgets were relevant to Long Wave single-mode links due to the impact of the long run of fiber optic glass itself and the associated connectors along the path. In data center short wave MM links, fiber loss is minimal and not normally a factor. Typically, most runs are not at the maximum distance. However, if that fiber run is at, or close to, its maximum distance of a 190 meter run at 8 Gb, or 125 meter run at 16 Gb, then adding fiber loss to the calculation might be relevant. The fiber loss would probably be in the .30 dB or less range, but something to consider for those very long high-speed MM runs or runs with many patch connections in between.

Link loss budget = TxMin - RxMin - Connector losses - Fiber loss

The minimum number of connectors is two (one at each end):

2 x 0.6 = 1.2 dB max

Having 0.6 dB loss per connector is the maximum. Typically, each connector has less than 0.3dB loss.

Historically, .75db was used with SC cables. Lucent, the original inventor of the LC connector, measured typical insertion loss in the

.20db range. There have since been LC connectors introduced with even better specs, in the .10dB or less range.

Using the above formula, using a typical 8 Gb SW SFP, the link budget equals:

[(-9) - (-11.7) - Connector losses] = (2.7 - Connector losses)

For example, if using quality connectors with high tolerances (i.e., low loss connectors, .20dB each) and four connections, one at each end and a patch panel in-between, then subtract another .80 and the result is 1.9.

Obviously, the optic transmitters will not be operating at the TxMin for all ports all the time, but for those runs at the maximum distance and speed with multiple connectors along the way, it is important to consider all the factors.

The new 16 Gb Connectrix B-Series switch and director products have a feature called D_Ports. With a switch at each end of the link, the links can be tested without the need for external testing equipment. This is on offline test for the ports being tested, but some customers are using these switches for this exact purpose during a DC build-out with a new fiber plant. This is an easy way for customers to verify the quality of the links that were just installed by their cable contractor.

Most of the major fiber optic cable manufacturers produce their own brand of fiber cabling that exceeds the typical OM4 distances. They do this by strictly controlling the quality and clarity of the glass to minimize modal dispersion and chromatic dispersion down the length of the fiber.

This is where a cable plant contractor can provide valuable input with regards to the quality of the connectors and fiber they' will be installing. If they are using an enhanced OM4 fiber and ultra low-loss connectors, their loss budget may be higher.

There are probably numerous SAN Administrators and support personnel who have seen first- hand the results of connecting higher speed devices or switches to old or suspect cable infrastructure. Consider the following scenario:

Typically, the switch optic is blamed (since that is where they see the CRC counts being incremented up by the switch), the optic is replaced, and all is back to normal, at least temporarily. However, in effect all that was done was the link was reset by the physical disconnection/reconnection and the CRC counts started counting

upward over time all over again. In actuality, the cable/connector is either just simply dirty, or the cable type itself really cannot handle the higher bandwidth at the distance of that run, or some combination of the two. Eventually the switch optic is blamed again.

This scenario repeated itself so many times across so many customers that EMC Customer Service procured and distributed optics cleaning kits for the field teams to use instead of just replacing optics. For EMC Service Personnel that want more information about this cleaning kit, see the following EMCU training material:

Course Title: *Inspect before you Connect - An Introduction to Fibre Optic Inspection and Cleaning*

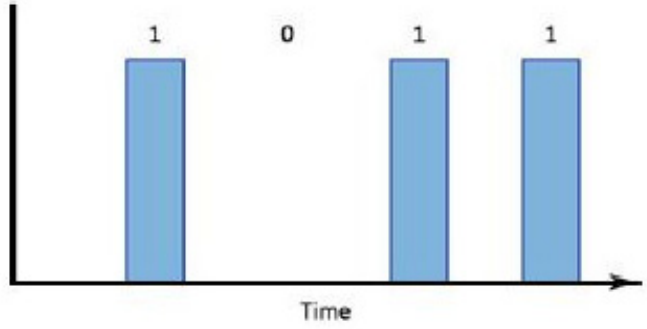
Course Number: MR-5WN-FOINSPECT

One of the simplest visual depictions of what is actually occurring at these higher speeds is from Panduit. The following descriptions of modal and chromatic dispersion and graphics show the impact of speed to light signals. Add dirt, dust, scratches, or old lower bandwidth fiber into the mix and it is easy to see just how quickly things can deteriorate due to a suspect physical layer.

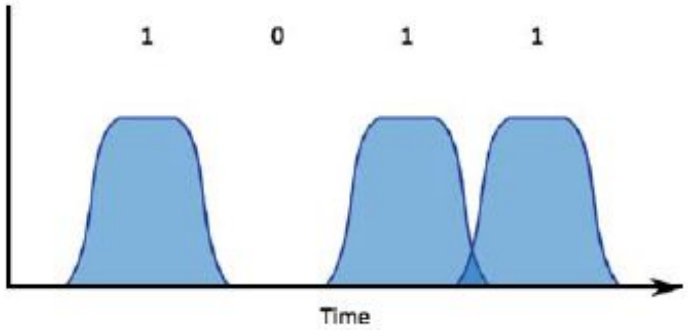
Modal dispersion is the broadening of an input signal as the optical power of the signal is split into different optical paths, or modes, in the core of the fiber, and each mode travels down the length of the fiber at different speeds.

Chromatic dispersion is a broadening of the input signal as it travels down the length of the fiber caused by the fact that different wavelengths of laser light travel at different speeds.

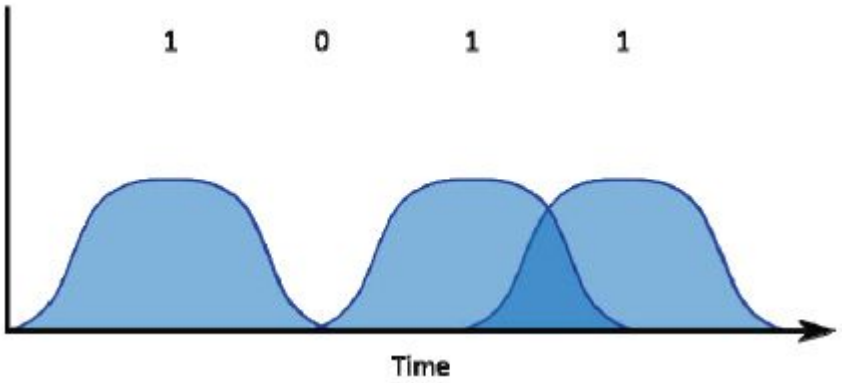
The following graph represents a stream of 1s and 0s as it would enter a multimode fiber:



When the pulses emerge from the fiber at the other end, they are distorted as a result of modal and chromatic dispersion.



Eventually, as the distance lengthens or the speed of the pulses increase, e.g., 10 G to 40 G Ethernet, the effects of dispersion make it difficult and nearly impossible to distinguish the individual pulses.



The pulse spreading is caused by both modal and chromatic distortion and is one of the main factors that limits the data rate and reach of a multimode fiber system.

Although modal dispersion has historically been more of an impact on performance than chromatic dispersion, improvements in the fiber manufacturing process have reduced its impact, leaving chromatic dispersion as the most important distortion impacting performance at higher data rates. The effect of chromatic distortion becomes more and more of a problem as the data rate increases and the reach becomes longer and longer. This is one of the reasons why chromatic distortion was not an issue in the past: the speeds were slow enough so that it had negligible impact. This is not the case today with 10 G, 40 G, and 100 G Ethernet, and 8 G and 16 G Fibre Channel.

Data transfer rates

The rate at which data will be transferred across a given medium is dependent on a number of factors. This section will consider the medium type first. In the case studies that follow, almost all of the implementations will use a serial communication protocol over an optical medium (i.e., Fibre Channel over multimode fiber).

The next factor to be considered is the signaling rate. In these examples, the signaling rate will equal the bit rate, but this is not always the case in serial communication protocols. Table 17 compares Fibre Channel to SONET and SDH in terms of their bit rates.

Table 17 Bit rates

FC abbreviation	MB/s definition	MB/s max theoretical	Signal rate MBd	Bits per second (bps)	Signal rate Mbps	Mb/s max theoretical	Optical Carrier Level	SONET Frame format	SDH level and Frame format
				51840000	51.84	48.69	OC-1	STS-1	STM-0
				155520000	155.52	150.36	OC-3	STS-3	STM-1
				622080000	622.08	601.344	OC-12	STS-12	STM-4
1 Gb/s	100	103.06	1,062 5	1062500000					
				12441600000	1244.16	1,202 668	OC-24	STS-24	STM-8
2 Gb/s	200	206.13	2,125	2125000000					
				2488320000	2488.32	2,405 376	OC-48	STS-48	STM-16
4 Gb/s	400	412.25	4,250	4250000000					
8 Gb/s	800	825	8,5	8500000000					
				9953280000	9953.28	9,621 504	OC-192	STS-192	STM-64
10 Gb/s	1200	1236.63	10,518 75	10518750000					
16 Gb/s	1600	1642	14,025	14025000000					
				39813120000	39813.12	38,486 016	OC-768	STS-768	STM-256

As shown in Table 17 on page 135, Fibre Channel environments can currently operate at 1 of 4 signaling rates (shown in megabaud MBd in the table) or data rates (also known as link speed, shown in MB/s in the table). In order to convert the link speed to a usable number in MB/s, you need to consider that the maximum data rate in MB/s will be decreased by protocol overhead, such as 8b/10b encoding, Start of Frame delimiter (SOF), Frame Header, CRC, End of Frame delimiter (EOF), and at least six primitives between the EOF of one frame and the SOF of another. The following formula can be used to determine maximum throughput.

Maximum throughput in MB/s = (((bit rate*.8)*.97)/8)/1000000

To calculate this formula:

1. Take the bit rate and multiply it by 0.8 to remove the 8b/10b overhead.
2. Take the product from Step 1 and remove the minimum overhead caused by the SOF (4-bytes), Frame Header (24-bytes), CRC (4-bytes), EOF (4-bytes) and the six primitive signals (24-bytes) that are required between each EOF and SOF. This overhead works out to be 3% (actually 3.4%) when frames are 2048 bytes long (using 2048/(2048+4+24+4+4+24) it works out to 0.966).

   To remove this minimum overhead, multiply the product from Step 1 by 0.97.
3. Take the product from Step 2 and divide it by 8 to get the number of bytes per second.
4. Take the quotient from Step 3 and divide it by 1000000 to get MB/s. This is the number that is displayed in the "MB/s max theoretical" column in Table 17.

Exception

The exception to the above rule is the 10 Gb/s link. 10 Gb/s uses 64b/66b encoding and this slightly changes the math:

Maximum throughput in MB/s = (((bit rate* .97)*.97)/8)/1000000

Note that the only difference is that the 0.8 was removed and replaced with 0.97. The 0.97 was obtained by division of (64/66).

The maximum data rate is also affected by the amount of buffers that a receiver has granted to a transmitter. In environments where there are no long distance links, having a large number of BB_Credits is not as beneficial as one might think.

For environments that do have long distance links, the number of BB_Credits available can have a dramatic impact on the maximum data rate. See Figure 53 and Figure 54 on page 138.

For those of you more comfortable with Mathcad, the calculations for 16 G FC are shown in Figure 52. Note that 16 G uses 64b/66b.

Signaling rates:
4Gb/s = 4.25 * 10^9
8Gb/s = 8.5 * 10^9
10Gb/s (Ethernet) = 10.3125 * 10^9
10Gb/s (FC) = 10.51875 * 10^9
16Gb/s = 14.025 * 10^9

$$S_{SignalRate} := 14.025.10^{9}$$

$$S_{uncoded} := S_{SignalRate} \cdot \left(\frac{64}{66}\right) = 1.36 \times 10^{10}$$

$$S_{noOverhead} := S_{uncoded} .966 = 1.314 \times 10^{10}$$

$$B_{bytessec} := \frac{S_{noOverhead}}{8} = 1.642 \times 10^{9}$$

$$D_{MBsRaw} := \frac{B_{bytessec}}{10^{6}} = 1.642 \times 10^{3}$$

Figure 52 Matchcad calculations for link speed

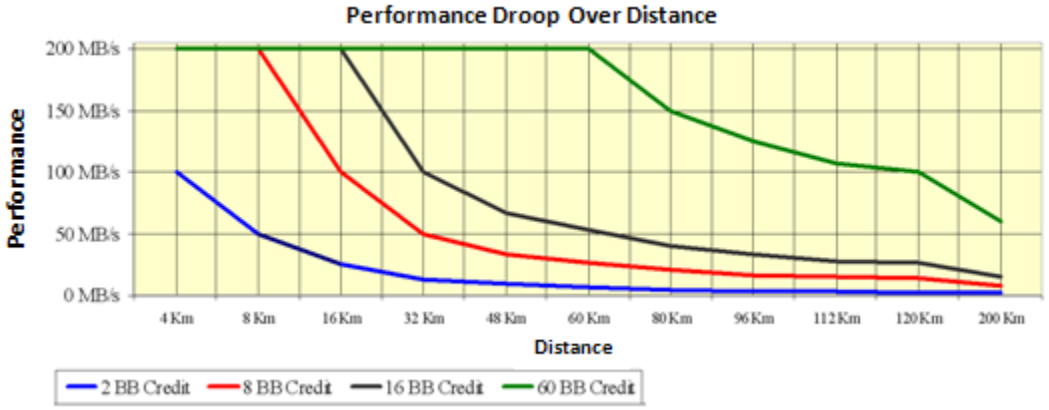


Figure 53 **200 MB/s link with max 60 BB_Credits**

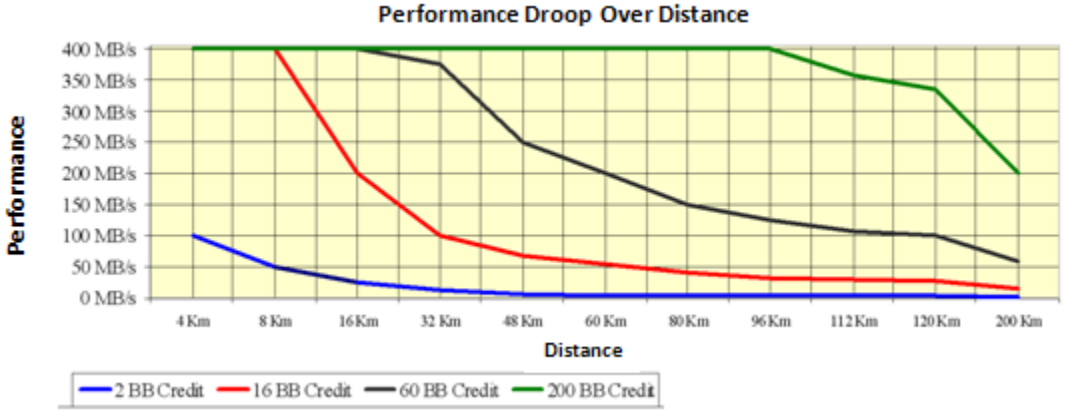


Figure 54 **400 MB/s link with maximum of 200 BB_Credit**

If the environment under consideration will experience a significant performance drop due to the amount of distance involved, distance extension solutions are available to help boost the performance. Refer to the *Extended Distance Technologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab, for more information.

Another factor that can significantly impact performance is the number of bit errors on a link. For more information, refer to "Congestion and backpressure" on page 217.

Fibre Channel port types

This section discusses types of Fibre Channel ports common to all vendors and then vendor-specific port types.

Standard Fibre Channel port types

There are several types of Fibre Channel ports, some of which are shown in Figure 55 and listed in Table 18.

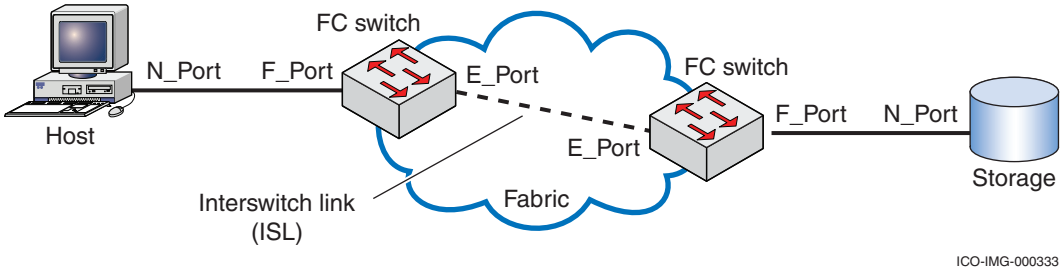


Figure 55 Standard Fibre Channel port types

Table 18 Standard Fibre Channel port types (page 1 of 2)

Port type	Description
N_Port	A port on a node outside the fabric
NL_Port	An N_Port that can also connect to a Fibre Channel arbitrated loop
F_Port	A port, located on a switching device, that connects to an N_Port and brings that connection (internally within the switch) to the fabric
FL_Port	An F_Port that can also connect to a Fibre Channel arbitrated loop
E_Port	A port, located on a switching device, that connects to an E_Port on a different switching device over an interswitch link (ISL)

Table 18 Standard Fibre Channel port types (page 2 of 2)

Port type	Description
G_Port	A port that can act as either an E_Port or an F_Port
GL_Port	A port that can act as E_Port, an F_Port, or an FL_Port

Vendor-specific Fibre Channel port types

In addition to the standard Fibre Channel port types each platform supports, they may also support some vendor-specific port types that provide enhanced functionality beyond what is defined in the standard. To determine which port types are supported by a specific platform, see the hardware section for that platform in the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab. For a list of all port types supported by each platform vendor, see the following vendor sections.

Brocade

The following port types are supported by Brocade:

VE_Port: An FCIP port on an FC switch will create a Virtual E_Port. This is physically an IP/Ethernet interface, but each FCIP tunnel "looks" like an FC E_Port to the rest of the fabric.

EX_Port: FC Routers use EX_Ports instead of E_Ports on routed interfaces. To connect a router to a switch, you connect its EX_Port to another switch's E_Port. The link formed is referred to as an IFL.

VEX_Port: A port that combines both FCIP and FC Routing features to allow the creation of a Virtual EX_Port.

IFL: The connection between an E_Port and an EX_Port is an Inter-Fabric Link.

Cisco The following port types are supported by Cisco:

TE_Port:	Trunking port (VSAN Trunking port). This port allows a single ISL or Port Channel to transport I/O for multiple VSANs.
SD_Port:	Spanning destination port. When troubleshooting a connectivity problem, it is frequently necessary to capture a trace between the FC switch and the device being attached. This can be accomplished by inserting an analyzer inline between the device and the switch or by mirroring (spanning) the frames between the two ports to another port called the SD_Port or Spanning destination port.

Fibre Channel Arbitrated Loop (FC-AL)

When Fibre Channel was first introduced, it was a new technology and everything was expensive. Switches, hubs, and node transceivers proved to be costly. Arbitrated Loop topology lies between point-to-point and switched fabric in that it provides more connectivity than point-to-point with up to 126 NL_Ports in a loop, but less than switched fabric, which has the ability in theory to support up to 16 million ports. It was a cost-effective way of connecting a limited number of ports in a loop single network.

This is considered an older technology, although it is still used.

Hubs

Fibre Channel hubs are used with Fibre Channel Arbitrated Loop (FC-AL) to increase server and storage connectivity. Using a hub, multiple servers can access multiple storage devices.

Note: EMC Symmetrix storage systems are qualified with hubs in limited configurations with HP-UX, Sun Solaris, Windows NT, Windows 2000, Windows 2003, and Siemens servers.

FC-SW (Fibre Channel switched fabric)

Fibre Channel switched fabric (FC-SW) is one or more dynamic Fibre Channel switches connecting multiple devices (see Figure 56 on page 144). FC-SW involves a switching device (the fabric) interconnecting two or more nodes. Frames are routed between source and destination by the fabric.

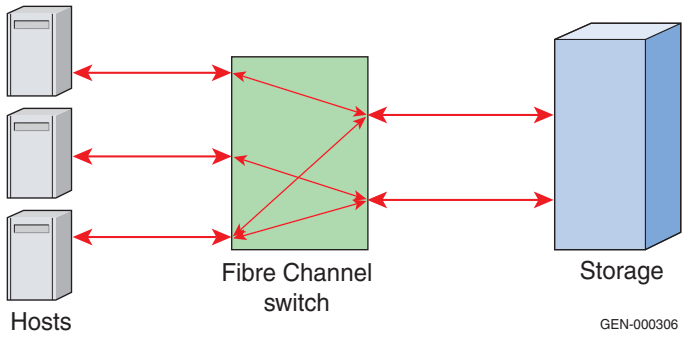


Figure 56 Switched Fabric example

Note: A *node* is any hardware device connected to one or more other hardware devices over Fibre Channel. Each node has at least one port that connects to ports in other nodes.

This section discusses:

FC-SW terminology

This section discusses terms used in FC-SW.

Embedded port

Every FC switch needs to process incoming frames which are destined for a well known address such as FFFFFE (Login Server). Some switch implementations send all requests destined for a well-known address to a special internal port, known as the *embedded port*.

PLOGI

PLOGI, or Port Login, is one port's way of requesting permission to login to another port. With the exception of SCR (State Change Registration), PLOGI needs to be sent and accepted before any additional requests are made between two N_Ports. For more information, refer to "N_Port Login (PLOGI)" on page 270.

Name Server

Formally known as Distributed Name Server, the Name Server, defined in FC-SW-4 (Section 9.3) is responsible for name registration and management of Nx_Ports that are attached to the switch in which it resides. These entries are stored in a locally resident database. Each switch exchanges its name server information with other switches in the fabric to maintain a synchronized, distributed name service.

These exchanges of information may occur periodically but are usually triggered by a fabric-wide SW_RSCN (switch registered state change notification). Each switch in the fabric must distribute SW_RSCNs throughout the fabric whenever a change takes place in its local name server database. The exchange of information occurs as F-Class traffic.

Fabric controller

Each switch has a local entity called the Fabric controller. The Fabric controller is defined in FC-SW-4 and is responsible for managing and distributing RSCNs to those Nx_Ports that register for them. The Fabric controller is responsible only for the notification that a change has occurred; it is the responsibility of the Nx_Port to re-query the Name Server to identify what changes have occurred. The Fabric controller is also used extensively during zoning changes and the Build Fabric process, including the assignment of Domain IDs.

Domain IDs

A Domain ID is a unique identification number provided to each switch in the fabric. The Domain ID and the port number associated with an Nx_Port's connection to the fabric are unique parts of the 24-bit Fibre Channel address for the Nx_Port. Each switch can be

assigned a preferred Domain ID, which it uses if there are no Domain ID conflicts in the current fabric.

On some platforms, each preferred Domain ID can also be set to be *persistent*. When Domain ID persistence is set, a switch will not join the fabric unless it can obtain its preferred Domain ID.

When merging two fabrics, if the same Domain ID is being used by multiple switches, a Domain ID overlap is said to exist and the switches being joined will segment from each other. When switches segment, no class F traffic or user data passes between them and the fabrics remain separate. It is possible to allow disruptive rebuilds to automatically take place when a Domain ID overlap exists, but this is not recommended since the Nx_Ports on the switch with the altered Domain ID will end up with different 24-bit FC IDs. The addition of a switch with a unique Domain ID to a fabric causes a *build* fabric. The disruptive rebuild to fix the overlap condition is referred to as a *reconfigure fabric*.

Domain ID negotiation

EMC recommends that you set all switches to have a unique persistent preferred Domain ID, to limit the chances of incurring problems resulting from duplicate Domain IDs. An example of a problem that can occur if unique persistent preferred Domain IDs are not used is following a power event. If not all of the switches come up in a timely manner, two different fabrics may form, each using the same set of Domain IDs. Unique persistent preferred Domain IDs can also shorten the time it takes for a switch to join/rejoin the fabric, because less negotiation is necessary when the fabric is rebuilding.

Since the Domain ID is used in the addressing of the components on the switch, Domain IDs cannot be changed unless the switch is taken offline first. Taking a switch that has a Domain ID conflict offline and then setting it back online will allow it to automatically negotiate with the principal switch for a new unique Domain ID as long as the Domain ID is not persistent. In this instance it will not be able to use its preferred Domain ID. For this reason even if switches are not currently in the same fabric, EMC recommends that all switches under your administration be given unique Domain IDs. This will allow you to merge them into a fabric at a later date without any manual intervention.

Principal switch placement and negotiation

One switch in the fabric is responsible for the distribution of Domain IDs, and plays a role in the route creation for fabric management

traffic. This switch is known as the *principal switch*. Since this switch must communicate with all other switches and is the basis for fabric traffic routing, this switch should be centrally located. This will assist in the uniform delivery of information in the fabric, and provide a consistent response to fabric build events.

Principal switches are selected during both the creation of the fabric and during fabric reconstruction events. Two pieces of information located on the switch will determine the selection of the principal switch in the fabric.

The first piece of information examined during the selection process is the switch's principal switch priority setting. The switch management applications usually represent these priority settings as **Always**, **Default** and **Never**.

- A switch whose priority is set to **Always** will always be involved in the process of principal switch negotiation.
- A switch set to **Default** will continue to participate in the process if no other switch is set to **Always**.
- A switch set to **Never** will never participate in the principal switch selection process, even if there are no other switches participating.
- A fabric that has no switch capable of participating in the principal switch negotiation will segment.
- If several switches are set to always participate in the principal switch selection, the switch in this group with the lowest World Wide Name (WWN) will become the principal switch.
- If no switches are set to always participate, those switches that are set to **Default** will complete the negotiation for principal switch. The principal switch in this scenario will also be the switch that currently has the lowest WWN.

Connectrix M Series switches have the ability to set a switch priority. EMC recommends that you set selected M Series switches to **Always** based on their location in the fabric. To configure a Brocade director switch in the core of a fabric as the *principal* switch, issue the **fabricprincipal** command on the switch CLI running FOS 4.x and higher. This setting becomes active on the next reboot, or after the Build Fabric (BF) event.

FSPF Fabric Shortest Path First (FSPF) is an algorithm used for routing traffic. This means that, between the source and destination, only the

paths that have the least physical hops (described under "Hops" on page 149) will be used for frame delivery.

ISL An interswitch link (ISL) is a link between any two switches in the fabric. ISLs are used to pass both data and management traffic simultaneously. Since each Fibre Channel exchange is unique, there is no limitation on the types of data (RDF, backup, management, or disk traffic) that can be transferred over an ISL simultaneously.

Principal ISL

Principal ISLs construct the path from any switch to and from the principal switch. These ISLs are assigned during the creation of the fabric, and are not necessarily the shortest path between any switch and the principal switch. The algorithm for assigning principal ISLs creates only a single path between any switch and the principal switch. Failure of the principal ISL would cause a fabric event that would attempt to calculate a new principal ISL between the effected switch and the principal switch.

Primary paths

Primary ISL paths are those that are currently the Fabric Shortest Path First (SFPF) routes between a specific host and its storage, and are being used for data traffic. The failure of all primary ISL paths would cause data traffic to transfer to the available secondary paths that were now the SFPF routes between the hosts and storage. The link table and routing table would also be updated to reflect the changes in the topology.

Secondary paths

Secondary ISL paths are paths between a specific host and storage pair that are not currently the FSPF routes between them and, therefore, not being used for data traffic. A secondary ISL path may be *hot* in the sense that it is acting as a primary ISL path for other server/storage traffic or *cold*, indicating that it is currently not being used for any subscriber data traffic. While EMC recommends that secondary paths to data be provided in the fabric design, the number of cold paths should be evaluated for their opportunity costs, as well as for their possible usage in your sparing model.

Upstream and downstream ISLs

This is a principal ISL from itself to the principal switch. This principal ISL is also called the upstream ISL. All fabric F-Class traffic that needs to travel from a switch to the principal switch will travel this same path. Traffic that travels from the principal switch to another switch in the fabric will traverse the downstream ISL.

Hops

A hop is the transition between any two ports on the fabric. The number of hops in a path is the number of ISLs that must be traversed to get from the input port to the respective output port. EMC recommends minimizing the number of hops in the fabric, as well as limiting the number of hops between any switch and the principal switch. This can help the fabric to stabilize more quickly after a fabric event by minimizing the time it takes to transmit fabric change events across the fabric.

Routing

The routing algorithm for all switches uses Channels versus networks versus SANs. Every ISL (E_Port) that enters on one port will also be assigned an exit port for each neighboring domain (switch).

The route list is dynamic only in the sense that changes in the fabric topology will cause the routing table to be recalculated. Adding and removing ISLs, as well as adding or removing switches, requires the routing table on each switch to be recalculated. Only those frames that were initially routed to a missing ISL will have to be resent and rerouted if they were lost during the fabric event.

Without events in the fabric, both the routing table and the routing of data exchanges will be static. Only the process of trunking (described under "Trunking" on page 207) will dynamically route frames across ISLs based on traffic patterns and utilization.

Switches use a round-robin approach to loading traffic across multiple same-cost routes to the same destination.

Switches do not set preferences for ISL cables that are physically shorter than others. This means that no matter how much of a distance disparity there is between two ISLs traveling to the same destination, they will still receive the same routing preference (costing). This behavior may be noticeable only when you are working with great disparities in length between similar routes. An example of when this may occur is in a fabric connected by Dense Wave Division Multiplexing (DWDM), or other link-extending equipment. DWDM links usually occur in pairs, and it is common practice to have each link travel a unique physical path to the destination.

While having the links travel in different directions protects the environment from failures caused by physical incidents damaging all of the DWDM links at the same time, this method of cabling also leads to distance disparities between the links in the DWDM ring.

RSCNs Registered State Change Notifications (RSCNs) are usually sent out by the switch and are destined to an N_Port that has been impacted by a change in the fabric. Ports need to register for state change with the fabric controller in order to receive an RSCN.

SW-RSCNs Switch Registered State Change Notifications (SW-RSCNs) are when an N_Port logs into or out of a switch, the switch will send an SW_RSCN to every other domain in the fabric. The switch RSCNs, in combination with periodic GE_PTs between domains, keeps the distributed name server up-to-date on all switches in the fabric. When a switch receives an RSCN, it is supposed to generate an RSCN to any impacted device which has registered for State Change Notification.

NPIV N_Port ID Virtualization (NPIV) is when a Fibre Channel facility allows multiple N_Port IDs to share a single physical N_Port. NPIV allows multiple initiators (FC or virtual) to occupy a single physical N_Port. For more information, refer to "NPIV" on page 292.

Switches

One dictionary definition of a switch is *a device for making, breaking, or changing the connections in an electrical circuit*. This basic definition is helpful when working with any type of switch, whether it be for Fibre Channel or some other protocol, such as Ethernet. A switch simply makes, breaks, and changes connections. Figure 57 shows the construction of a basic switch.

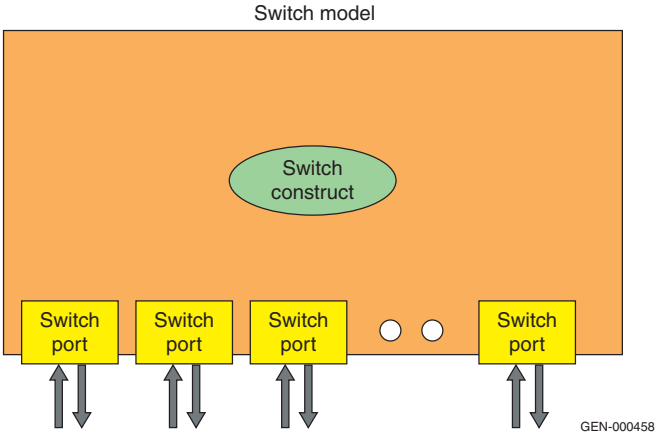


Figure 57 Generic switch construct

Although the physical implementations of switches made by each vendor are unique, they are all expected to provide a common set of services as defined in the Fibre Channel Fabric Generic Requirements Standards (also known as FC-FG, a copy of which can be found at http://www.t11.org). Some examples of these services are:

- Fabric F_Port/Login server
- Fabric controller
- Directory server
- Management server

Figure 58 shows a generic switch with services.

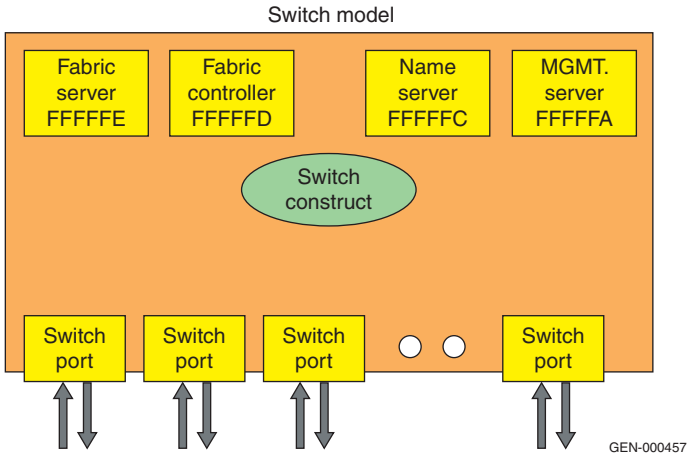


Figure 58 Generic switch construct with services

Every switch, regardless of manufacturer, is required to make these services available at certain *well-known addresses*, discussed next.

Well-known addresses

Each generic service defined in the Fibre Channel standards FC-PH has a special Destination ID (D_ID) associated with it. A D_ID is another term for a port address. The D_ID for these well-known addresses is in the FFFFF*x* format (where *x* is a value between 0 and F). Every switch, regardless of manufacturer, is expected to perform certain functions on frames that are sent to one of these well-known addresses. Frames that are destined for a well-known address are almost always forwarded to a special internal Fibre Channel port,

known as the Embedded port, making it easier for switch vendors to implement.

The fabric server

The fabric server is located at the well-known address of FFFFFE (see Figure 58). Although there are several possible requests that could be made to the fabric server, it is mainly used during the first part of the N_Port login process. The specific requests associated with this are actually the extended link service request FLOGI (Fabric Login).

The fabric controller

The fabric controller is located at the well-known address FFFFFD (see Figure 58 on page 151). This address identifies has special usage depending on the originator. If the originator is an attached external N_Port or NL_Port (attached through an F_Port or FL_Port), then the destination of a frame sent to FFFFFD is the fabric controller of the local switch. If the originator is the fabric controller of the local switch, then the destination of a frame sent to FFFFFD through an ISL is the fabric controller of the remote switch at the other end of the ISL.

The fabric controller also provides services to both N and NL_Ports as well as other switches.

For N_Ports, the fabric controller is usually used for State Change Registration (SCR). When an N_Port successfully completes SCR, it is eligible to received Registered State Change Notifications (RSCNs).

RSCNs are generated by the fabric controller and sent to all affected ports (if registered to be notified) whenever there is a change to the environment.

The directory server (or name server)

The directory server, more commonly known as the name server, is located at the well-known address FFFFFC (see Figure 58 on page 151). It is also used during the N_Port initialization process when an N_Port wants to register its FC-4 type and then later query for other FC-4s that are available through zoning and currently logged into the switch.

Route and path selection

Route and path selection is not provided at a well-known address, but is a service provided by all switches. Frames are routed based on their D_ID. (A common misconceptions is that they are routed based on their WWN.) In general, frames are started down the proper route by the ASIC on the receiving port of the switch or by information

provided by the receiving port to some other component within the switch. This is also true in a multiswitch fabric.

Fabrics

A fabric is a collection of switches that have been linked together through a Fibre Channel cable, also known as an ISL (interswitch link). The term fabric is generally used to indicate a group of switches, although it is technically correct to refer to a single switch that is not connected to other switches as a *single switch fabric*. Each switch can effectively be thought of as its own domain and, in fact, each switch has a unique Domain ID associated with it. Figure 59 shows an example of a switched fabric.

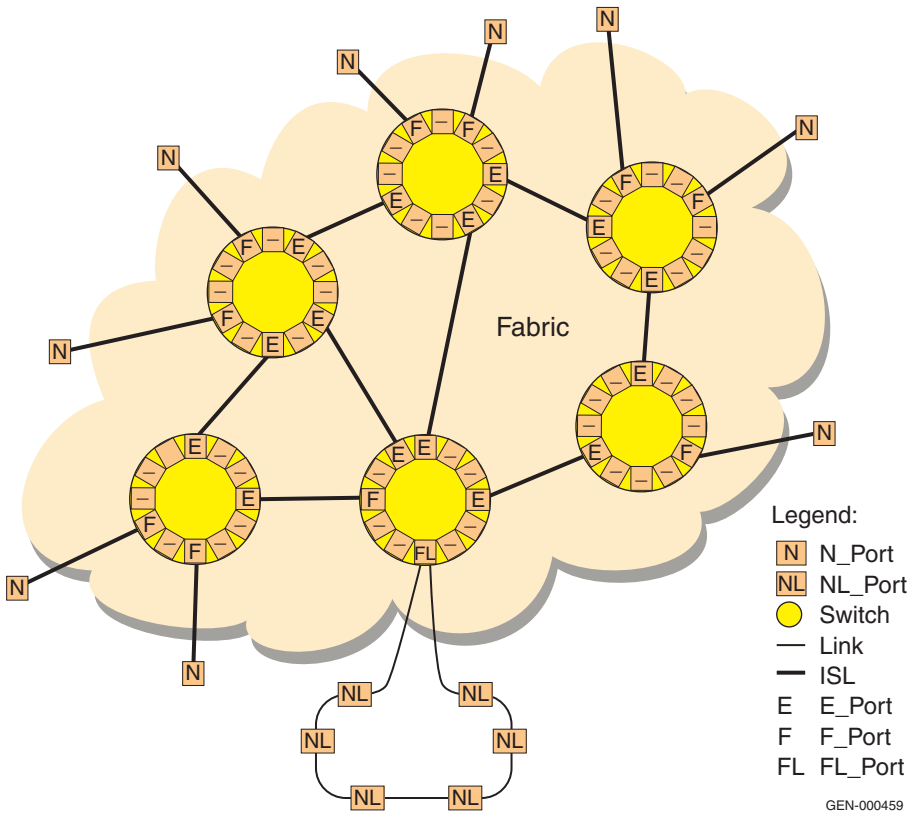


Figure 59 Switched fabric example

Build Fabric (Fabric Configuration) process

There are numerous books and reference materials available (including the *Fibre Channel Standard FC-SW*) that provide thorough explanations of the Build Fabric process (also referred to as Fabric Configuration process), down to the bit level. The information provided in this section will include only the major concepts of the Build Fabric process with the intention of both familiarizing the reader with the process of fabric creation and facilitating future investigation.

Topology

Throughout this section, the topology shown in Figure 60 on page 155 will be used to illustrate the Fabric Configuration process. The architecture is intended to maximize the readers' understanding of the process and is not intended to be an endorsement of the topology itself.

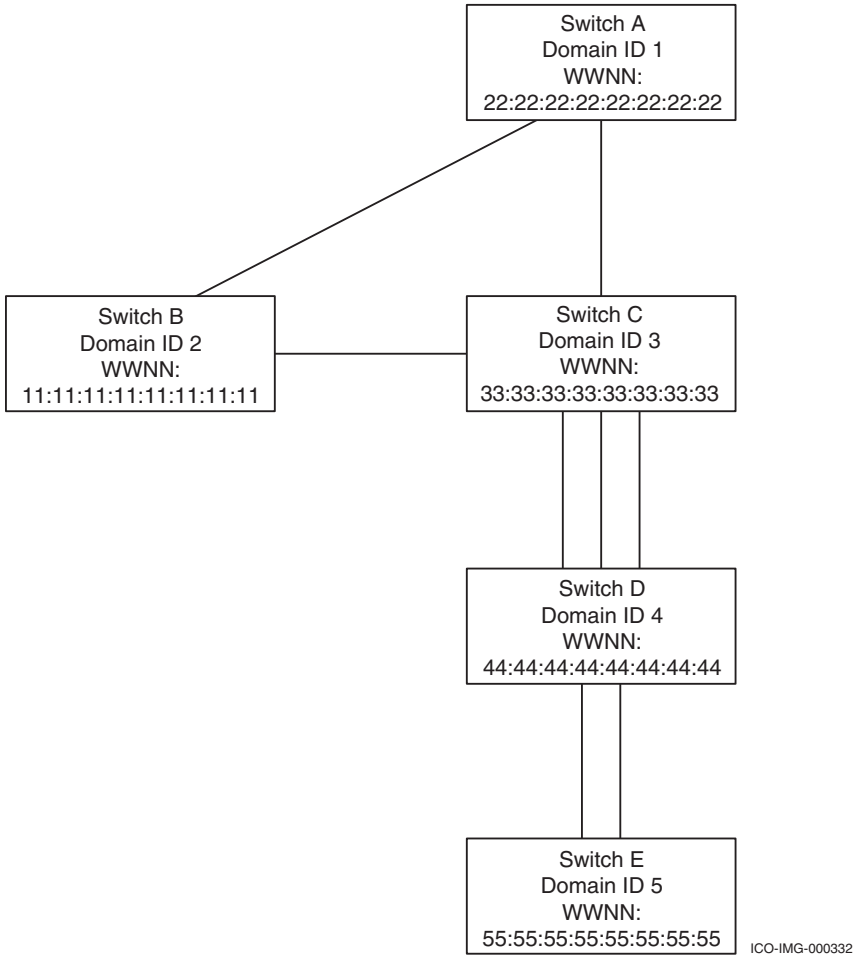


Figure 60 Build Fabric (Fabric Configuration) process

In this example, we will assume that a fiber cable has just been connected between switch D and E. The first step that must be taken is link initialization.

Negotiate port type

When two FC ports are connected, they will negotiate speed and bring the link up to the active state (Figure 61).

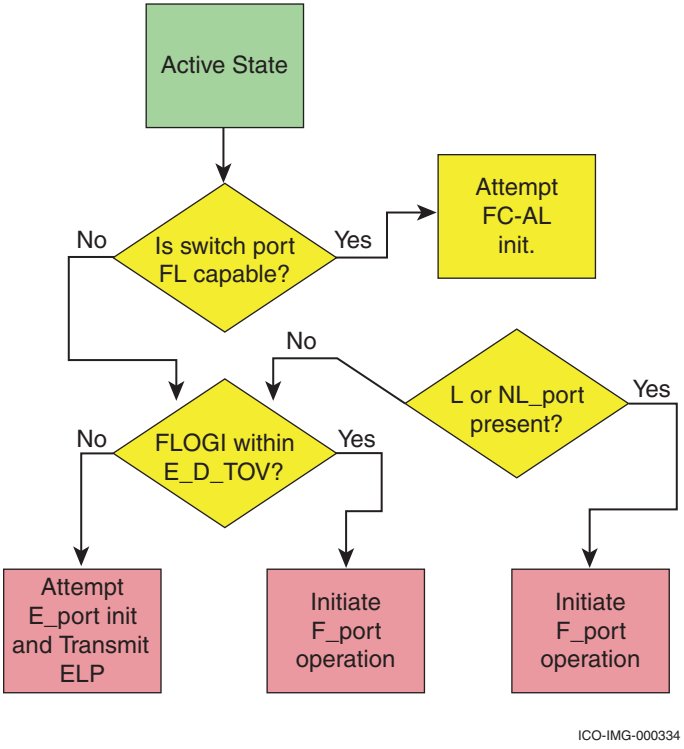


Figure 61 **Switch port initialization**

In the case of an HBA connecting to a switch, the HBA will send FLOGI, which indicates to the switch that a host is connecting to the switch. Therefore, the port on the switch will become an F_Port.

The same is not true when two switches are connected together. In this case, the switches usually wait a period of time for a FLOGI. If one is not received and the switch is configured to allow E_Port connections on the port that just came up, the switch will send out an ELP (Exchange Link Parameters) frame and try to initialize the port as an E_Port. If the ELP is accepted by the port on the other end of the link, then the switch will initialize the port as an E_Port and the Fabric Configuration process begins.

Fabric Configuration (Build Fabric) process

The purpose of the Fabric Configuration process is to allow for the formation of ISLs (Interswitch links) between different domains and to facilitate Nx_Port connectivity between them.

Note: For the sake of clarity, the word "domain" will be used in place of "switch" throughout this section. With the advent of VSANs and partitions, many domains can reside on the same switch physical switch.

The FC standard breaks up the Fabric Configuration (also referred to as the Build Fabric) process into the following logical chunks:

- Exchange Link Parameters
- Principal switch selection
- Domain ID acquisition
- Zoning Merge
- Path selection

For clarity, we will slightly reorganize them as follows:

- "ELP – Exchange Link Parameters" on page 158
- "Perform Link Reset (LR)" on page 161
- "ESC - Exchange Switch Capabilities" on page 161
- "EVFP – Exchange Virtual Fabric Parameters" on page 161
- "Principal switch selection" on page 163
- "EFP – Exchange Fabric Parameters" on page 163
- "Build Fabric (BF)" on page 164
- "EFP – Exchange Fabric Parameters (Selecting the principal switch)" on page 166
- "Domain ID acquisition" on page 170
- "Zoning merge" on page 174
- "Path selection and the FSPF protocol" on page 176
- "Assigning egress ports" on page 194

Each of these will be further discussed in this section.

Note: The frames that are sent between switches used a class of service called Class F. This means they all begin with the Start of Frame Fabric (SOFf) delimiter and are all Acknowledged by the receiving switch, like Class 2. For simplicity, the ACK frames have been left out of all the examples in this section, but it is important to realize that they are used throughout the process.

ELP – Exchange Link Parameters

Once the link has been initialized, the first frame that a switch will transmit is the ELP frame (refer to Figure 62 on page 160). The ELP is transmitted from one Fabric Controller to another (FFFFFD to FFFFFD) and contains the following information:

- R_A_TOV (Resource Allocation Time Out Value) — This value needs to be the same on both switches or the ISL will isolate (segment). This typically appears in the user interface as either an ELP failure or as incompatible flow control parameters.

  Note: Originally, the value of R_A_TOV was supposed to be equal to the value of E_D_TOV plus two times the maximum amount of time that a frame can exist in the fabric. However, today it is always left at a value of 10000 ms (10 seconds) and is not tied to how long a switch will hold onto a frame.

  It is important to note that the amount of time a frame can exist in the fabric depends on the number of hops between two NX_Ports (the number of ISLs that need to be crossed) and the value of the hold timer used by the switch being crossed. Currently, the hold timer value can vary from 500 ms up to 2000 ms. The number of hops can be up to three in EMC-supported configurations. See "FLOGI" on page 239 for more information.

- E_D_TOV (Error Detect Time Out Value) — This value needs to be the same on both switches or the ISL will isolate. This typically appears in the user interface as either an ELP failure or incompatible flow control parameters.

  The value of E_D_TOV is used for many different purposes, but in each case the E_D_TOV timeout value is used for detecting an error condition. An example that frequently occurs is when a port has zero transmit BB_Credit for E_D_TOV; the port is supposed to transmit the LR primitive to reset the BB_Credit counters to their login values.

- WWPN (World Wide Port Name) — The unique 64-bit address of the port transmitting the ELP.
- Switch name — The unique 64-bit address of the Domain to which the port transmitting the ELP belongs.
- Receive data field size — The maximum payload that is allowed by the switch.
- Flow control parameters:
  - Type of flow control being supported by the switch, i.e., R_RDY, VC_RDY.
  - Number of BB_Credits being granted by the switch — In configurations where distance is going to be a consideration (generally > 10 km), the BB value exchanged between the two FC ports has a significant impact on the maximum sustainable throughput rate.

Figure 62 shows the Exchange Link Parameters (ELP) process.

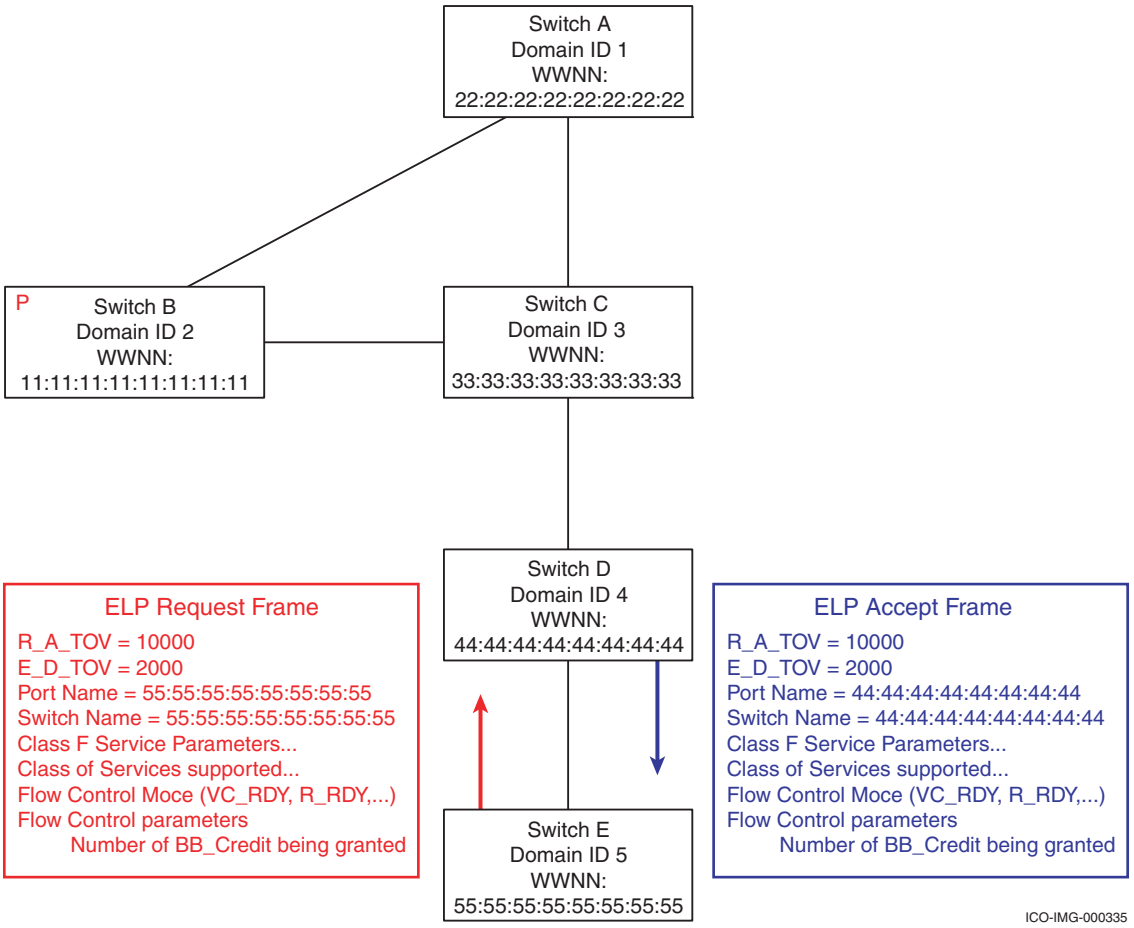


Figure 62 ELP Exchange Link Parameters process

If the port receiving the ELP at the other end of the link is a switch port and is configured to allow E_Ports, the ELP will be processed by the receiving switch. Depending on the switch vendor, features enabled, basic security settings (like Fabric Binding, Switch Binding, etc.), the ELP will either be *accepted* (if everything checks out) or *rejected* (if there is a mismatch that cannot be resolved).

If the ELP is accepted, then an ELP Accept frame is sent from the receiving switch back to the ELP originator. As with all of the Exchange requests, the contents of the ELP accept are identical in

type to the ELP request, except it is populated with information specific to the switch accepting the ELP.

If the ELP is rejected, both ports (one on each end of the physical link) should be shown as isolated (segmented).

Perform Link Reset (LR)

As mentioned in "ELP – Exchange Link Parameters" on page 158, two of the link parameters exchanged during ELP are:

- The type of flow control to be used
- The number of BB_Credits being granted by each switch port

Once this information has been exchanged, the Link Reset (LR) protocol is initiated to initialize each port to use these login values. The Link Reset protocol is kicked off by one switch transmitting Link Reset primitive sequences in place of IDLEs. When the other port receives and recognizes the LRs, it resets its credit value to the login value and starts transmitting Link Reset Response (LRR) primitive sequences instead of IDLEs. When the port transmitting LRs receives and recognizes the LRRs, it resets its credit value to the login value and begins to transmit IDLEs again. Once the IDLEs are received by the other port, it begins to transmit IDLEs and both ports are back at the active state. The Link Reset protocol can be invoked for many other reasons as well.

ESC - Exchange Switch Capabilities

After the ELP is completed, the switches have the option to exchange switch capabilities information. Outside of vendor-specific information, the primary use of the ESC request is to determine if the switches support virtual fabrics. If a switch supports virtual fabrics, it will list this capability in the ESC payload. If the responding switch also supports virtual fabrics, it will do the same. If both switches support virtual fabrics, the next frame to be transmitted by either switch will be the Exchange Virtual Fabric Parameters (EVFP). Otherwise they will continue on with the Exchange Fabric Parameters (EFP).

EVFP – Exchange Virtual Fabric Parameters

If both switches list support for virtual fabrics in their ESC/ESC accept, the EVFP will be used to share which virtual fabric features will be supported. The two features that will be discussed in this section are *Virtual Fabric Identifiers* and *Virtual Fabric Tagging*.

Currently there are two basic types of virtual fabric implementations: those that support virtual fabric tagging and those that do not.

An example of an implementation that supports virtual fabrics but does *not* support virtual fabric tagging is the Brocade ED-10000 (or Intrepid 10000). Starting with Firmware EOSn 9.0, users can configure each port on the Brocade ED-10000 to have its own unique Virtual Fabric Identifier or VF_ID. Ports in the same partition with the *same* VF_ID *can* all communicate with each other. Ports with *different* VF_IDs *cannot* communicate with each other. If two Brocade ED-10000s running EOSn 9.x or higher are connected to each other in an attempt to form an ISL, they will send EVFP. If they have the same VF_ID they will continue along the fabric configuration process. If they do not have the same VF_ID, they will segment from each other with a virtual fabric conflict.

One thing to note about this implementation is that for every VF_ID you want to have communicating between the two Brocade ED-10000s, you will need to have a separate ISL assigned exclusively to each VF_ID. They cannot share the ISL between the different VF_IDs since they do not support VF tagging so there is no way for the switch port at the other end of the ISL to determine which virtual fabric to route the incoming frame to. This is why virtual fabric tagging and the Virtual Fabric Tagging Header (VFT_Header) becomes important. Figure 63 shows a normal FC Frame and another that has the 8 byte VFT_Header prefix added to it.

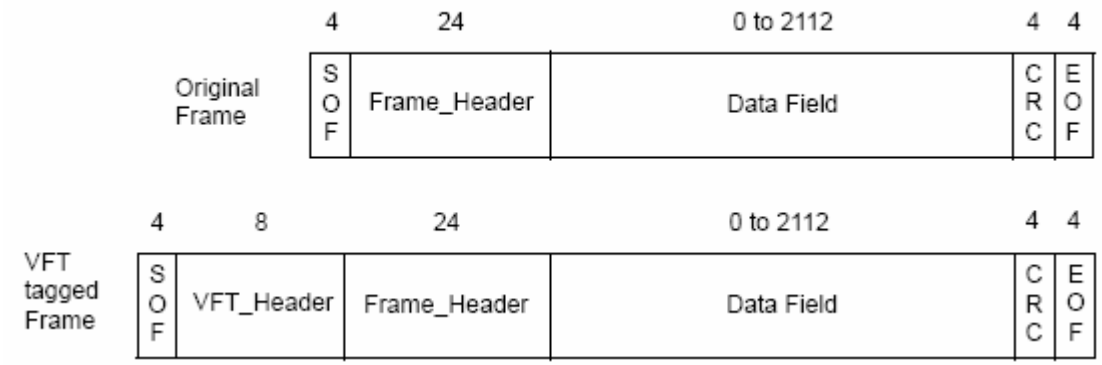


Figure 63 Normal FC Frame and a VFT_Header Frame

Implementations that use the VFT_Header can accommodate multiple virtual fabrics simultaneously using the same ISL. An example is Cisco and its VSAN trunking feature.

Once each switch has shared its supported virtual fabric features, the ISL will be in one of the following three states:

- **Isolated** — VF_ID is mismatched.
- **Active without VFT Headers** – Class F traffic is allowed to flow between the two domains and the Fabric Configuration process can continue.
- **Active with VFT Headers** (i.e., EISL) – Class F traffic from multiple VF_IDs (VSANs) are allowed and the Fabric Configuration processes can continue.

Principal switch selection

During the Fabric Configuration process, each fabric selects one switch to act as the principal switch. The role of the principal switch is to assign Domain IDs to the rest of the switches in the fabric. The principal switch selection process begins with the Exchange of Fabric Parameters (EFP).

EFP – Exchange Fabric Parameters

The EFP exchange is used several times during the Fabric Configuration process as follows:

- After an ISL has been established and before the Build Fabric request is sent out.

  Note: The purpose of this initial EFP is to ensure that a Domain ID overlap does not exist between the two fabrics being joined.

- After the Build Fabric is transmitted.
- After the principal switch is selected.
- After each switch is assigned a Domain ID.

This section covers the initial use of EFP. The subsequent EFPs are covered in the following sections: "Build Fabric (BF)" on page 164 and "EFP – Exchange Fabric Parameters (Selecting the principal switch)" on page 166.

The EFP request payload consists of the principal switch name (WWNN) and its priority as well as a list of Domain IDs already in use. The list of Domain IDs are each associated with a specific switch name. In this way, during the initial EFP, Domain ID overlaps can be detected by comparing the list of Domain IDs and ensuring that if the same Domain ID exists on both sides of the ISL, that it is associated with the same switch name. If the same Domain ID exists but is

associated with a different switch name, the ISL will isolate and no additional Class F traffic will be exchanged on that ISL, in other words, the Fabric Configuration process ceases on that ISL.

In Figure 64, since there is no Domain ID overlap, the Fabric Configuration process continues with Build Fabric.

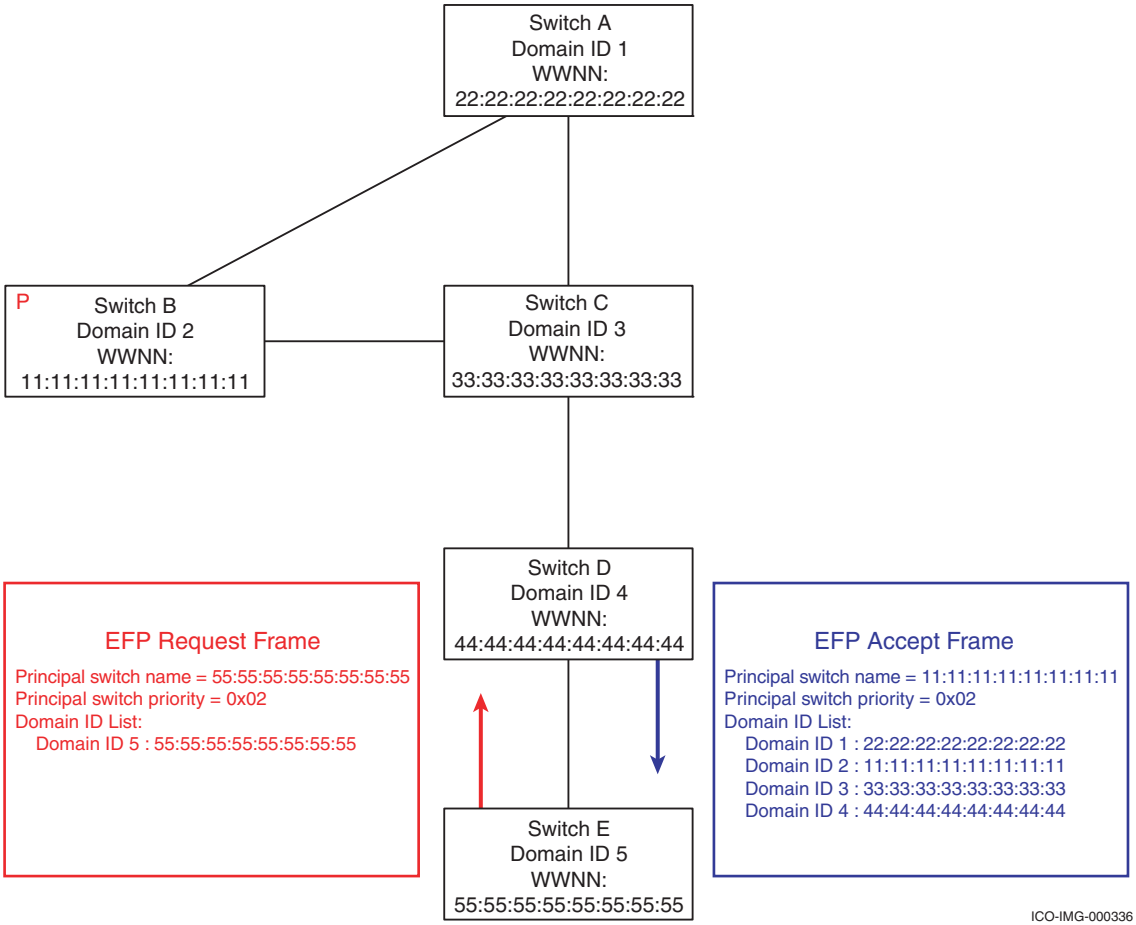


Figure 64 EFP request payload

Build Fabric (BF)

If the initial EFP has completed and has not resulted in an isolated ISL, the Fabric Configuration process continues with Build Fabric (BF) being transmitted on all E_Ports that have not already received a Build Fabric from the switch on the other end of the ISL. The Build

Fabric request payload consists of a single 4-byte word that contains the Build Fabric command itself.

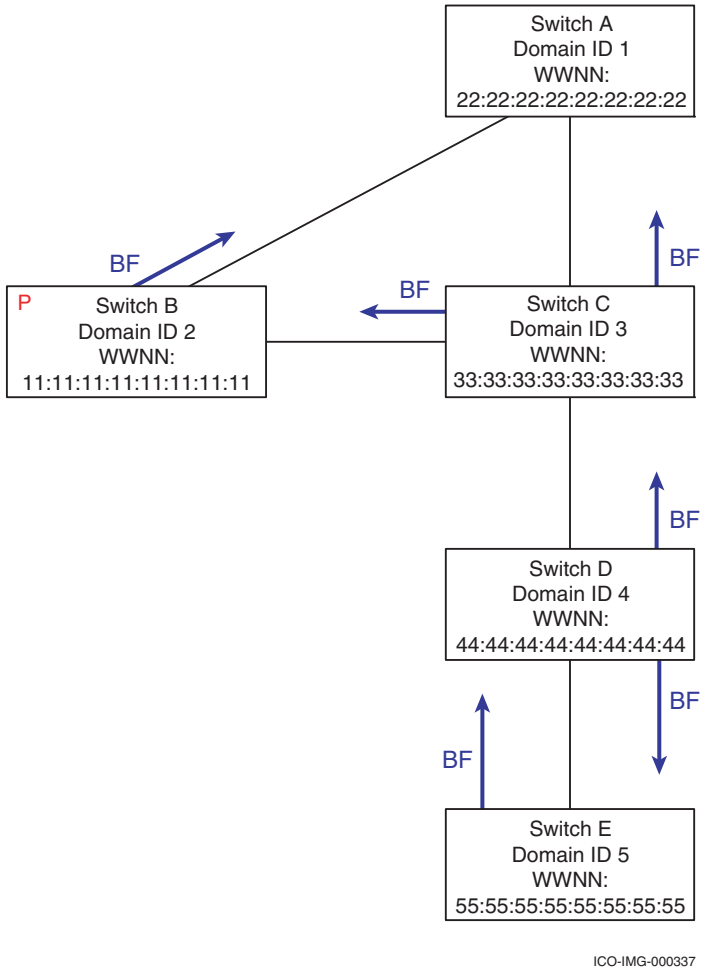


Figure 65 **Build fabric**

In Figure 65, BF is sent from switch D and E at the same time. Switch D sent the BF to both C and E and switch E sent the BF to switch D. All three switches should accept the BF from the other switches. When switch C receives the BF, it will accept it and then transmit BF out all E_Ports that it has not yet received a BF on. Finally switch B will transmit BF to switch A.

As soon as the switch transmits the BF, it must wait a period of time called F_S_TOV (Fabric Stability Time Out Value) before continuing on with the next phase of Fabric Configuration. F_S_TOV is defined as 5 seconds.

EFP – Exchange Fabric Parameters (Selecting the principal switch)

Once F_S_TOV has expired, the switch can start initiating and responding to EFP requests. These EFPs are used to select the principal switch. Many different starting conditions and ending configurations are discussed in the EFP section of the *Fibre Channel Standard FC-SW*. A few of the more important concepts are:

- When a switch initially transitions from offline to online, it does *not* have a Domain ID assigned to it. As a result, if this switch were connected to other switches in the fabric while it was coming online, it would send out an EFP containing a Domain ID list with zero entries in it. In this case, the switch coming online is not eligible to become the principal switch unless there is no switch in the fabric capable of becoming the principal. The switch coming online will take note of the first ISL that an EFP was received on and treat it as the principal ISL.
- When two switches are joined, and each has a non-zero Domain ID list as well as a switch priority other than 0xFF, both switches are candidates to become the principal switch. The principal switch is determined by each switch exchanging EFPs with every other switch it is attached to and initially comparing its own value of switch priority and switch name with the other switch to which it is attached.

  When a switch discovers another switch with a lower priority, or a switch with the same priority but a lower switch name, it starts sending out this other switch's priority and switch name instead of its own in subsequent EFPs. This process continues for a time equivalent to 2 x F_S_TOV. The switch that has retained its own switch priority and switch name at the end of 2 x F_S_TOV is declared to be the principal switch. During this process, whenever a switch discovers a switch priority / switch name combination lower than its own, it takes note of the port it was received on as a potential candidate for principal ISL. At the end of this process, 3 x F_S_TOV (15 seconds) has elapsed since the start of the Fabric Configuration process. The initial F_S_TOV

was after sending BF, while the second and third were during the principal switch selection process in which EFP was used for the second time.

Example of EFP usage during the principal switch selection process

After the BFs were sent and F_S_TOV expired, all switches in the fabric will begin exchanging EFPs. One of the rules for accepting a BF is that the switch will clear its Domain ID list and replace the principal switch priority and principal switch name fields with its own values in subsequent EFP requests or responses.

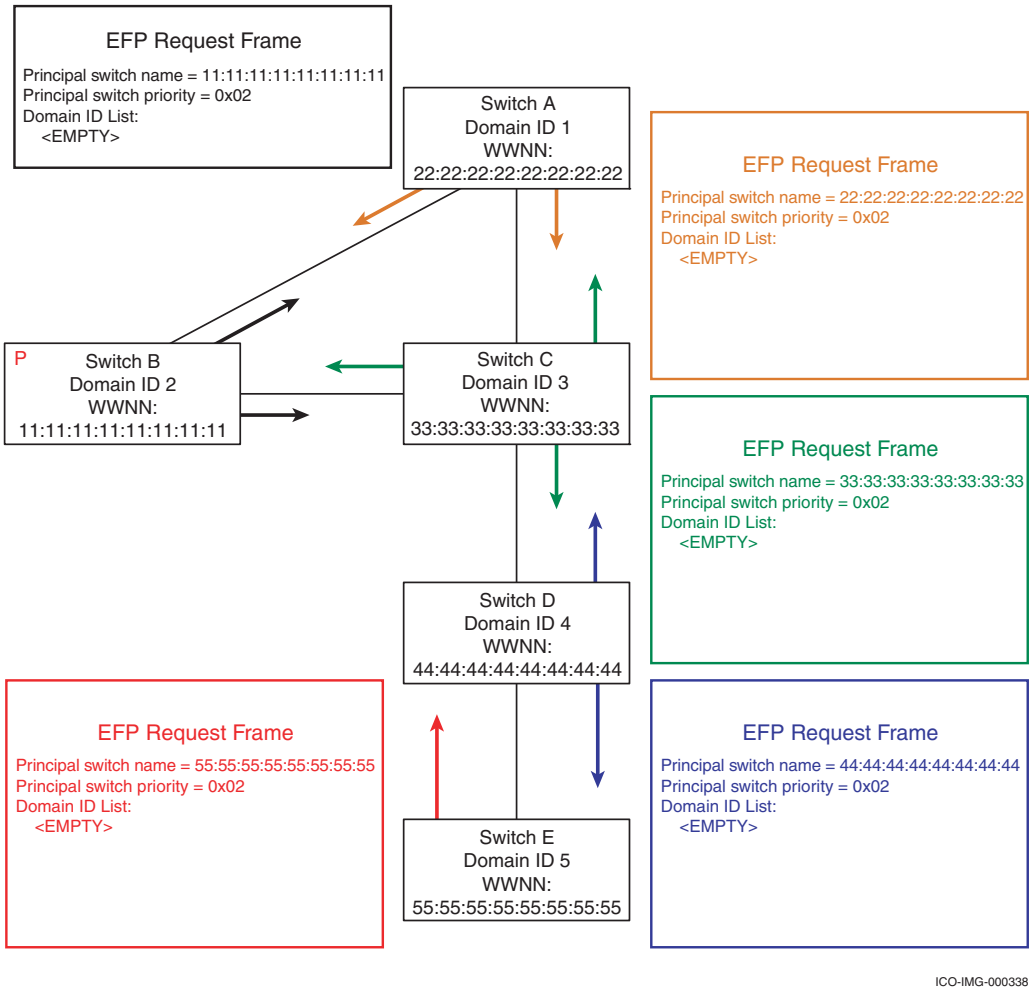


Figure 66 Principle switch selection process 1

In Figure 66 on page 167, the colors of the arrows illustrate what data is being sent in the EFP request. The EFP request is shown in the corresponding colored box.

As shown in Figure 67 on page 169, when switch B receives the EFP from switches A and C, it will compare their priority and principal switch name with its own. Since switch B's priority is the same, but its switch name is lower, switch B will continue to transmit EFPs and EFP accepts with its own switch name and priority.

When switch A or C receive the EFP from switch B, they will compare the priority and principal switch name with their own. Since switch B's priority is the same but switch name is lower than either switch A or C, they will replace their value of principal switch name and principal switch priority with that of switch B and transmit that value in subsequent EFP and EFP Accepts.

When switch D receives the EFP from switch C and E, it will notice that switch C's switch name is lower and begin to transmit switch C's switch name in subsequent EFP requests or accepts.

When switch E receives the EFP from switch D, it will begin to transmit switch D's switch name since it is lower than E's.

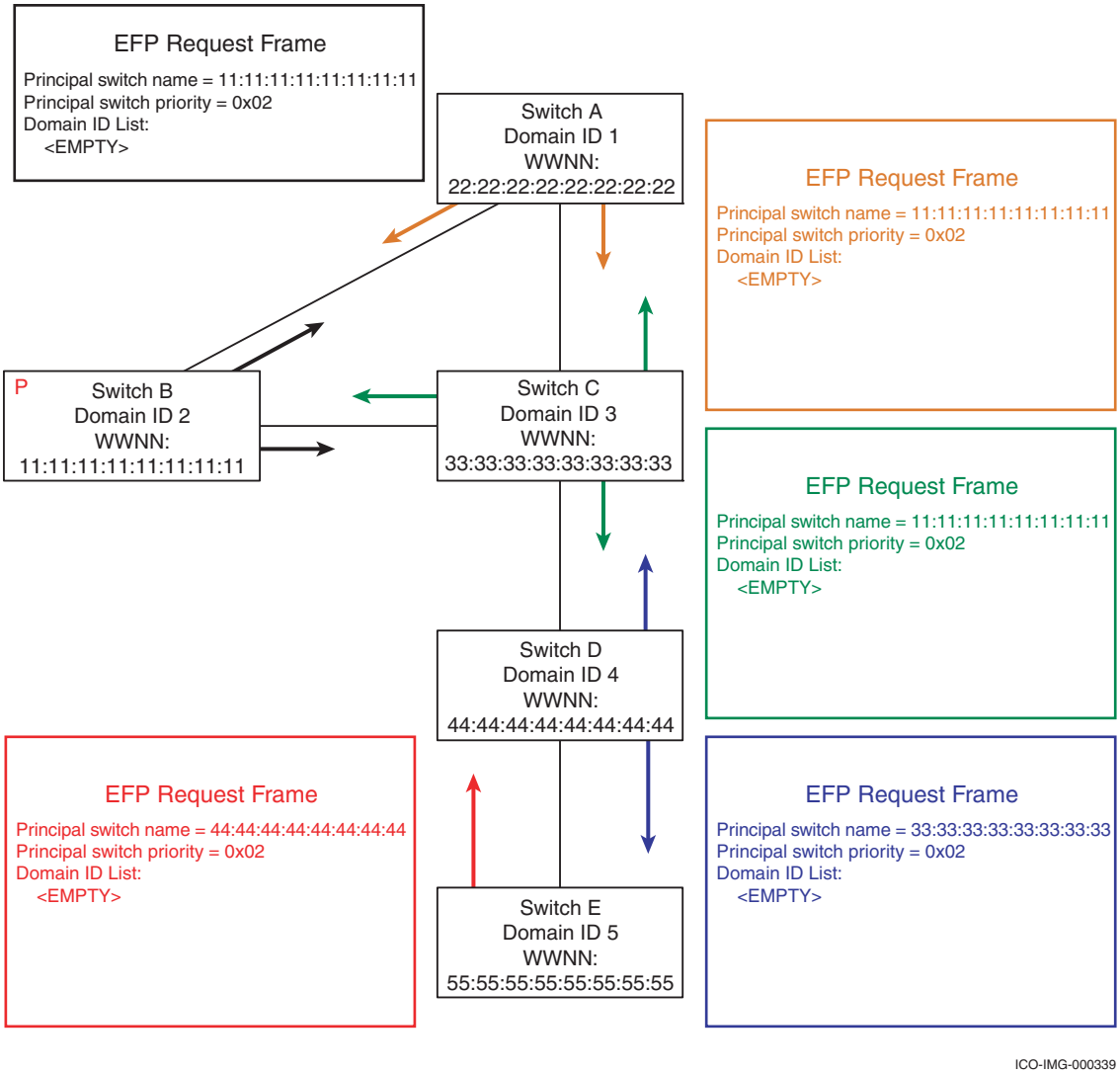


Figure 67 **Principle switch selection process 2**

Finally, switch D will receive the EFP that contains switch B's switch name and transmit it to switch E, which will then begin to transmit switch A's switch name.

After all the EFPs have been exchanged, all switches will agree that the switch with a switch name of 11:11:11:11:11:11:11:11 is the principal switch.

Domain ID acquisition

Once the principal switch has been selected, it will:

1. Assign itself a low switch priority value (0x02).
2. Clear its local copy of the Domain ID list.
3. Grant itself a Domain ID, (typically the last one it was using).
4. Transmit a Domain Identifier Assigned (DIA) on all E_Ports.

DIA - Domain Identifier Assigned

The DIA indicates that a principal switch has been selected and that the upstream switch has been assigned a Domain ID[1].

Once the principal switch has been selected and the appropriate amount of time has transpired, the principal switch will send out DIA on all E_Ports (refer to Figure 68 on page 171). Once the receiving switch accepts the DIA, it can send an Request Domain ID (RDI) to the principal switch via the principal upstream ISL. (The principal upstream ISL was determined when this switch received the EFP with the lowest switch priority / switch name.) The DIA request consists only of the switch name of the switch that initiated the DIA.

After sending the DIA, and before sending or responding to any EFPs, the principal switch will assign itself a Domain ID, typically the one it had previously. Once this is done, the principal will generate an EFP to inform the other switches in the fabric of the new switch in the fabric.

1. FC-SW-4 section 6.1.6

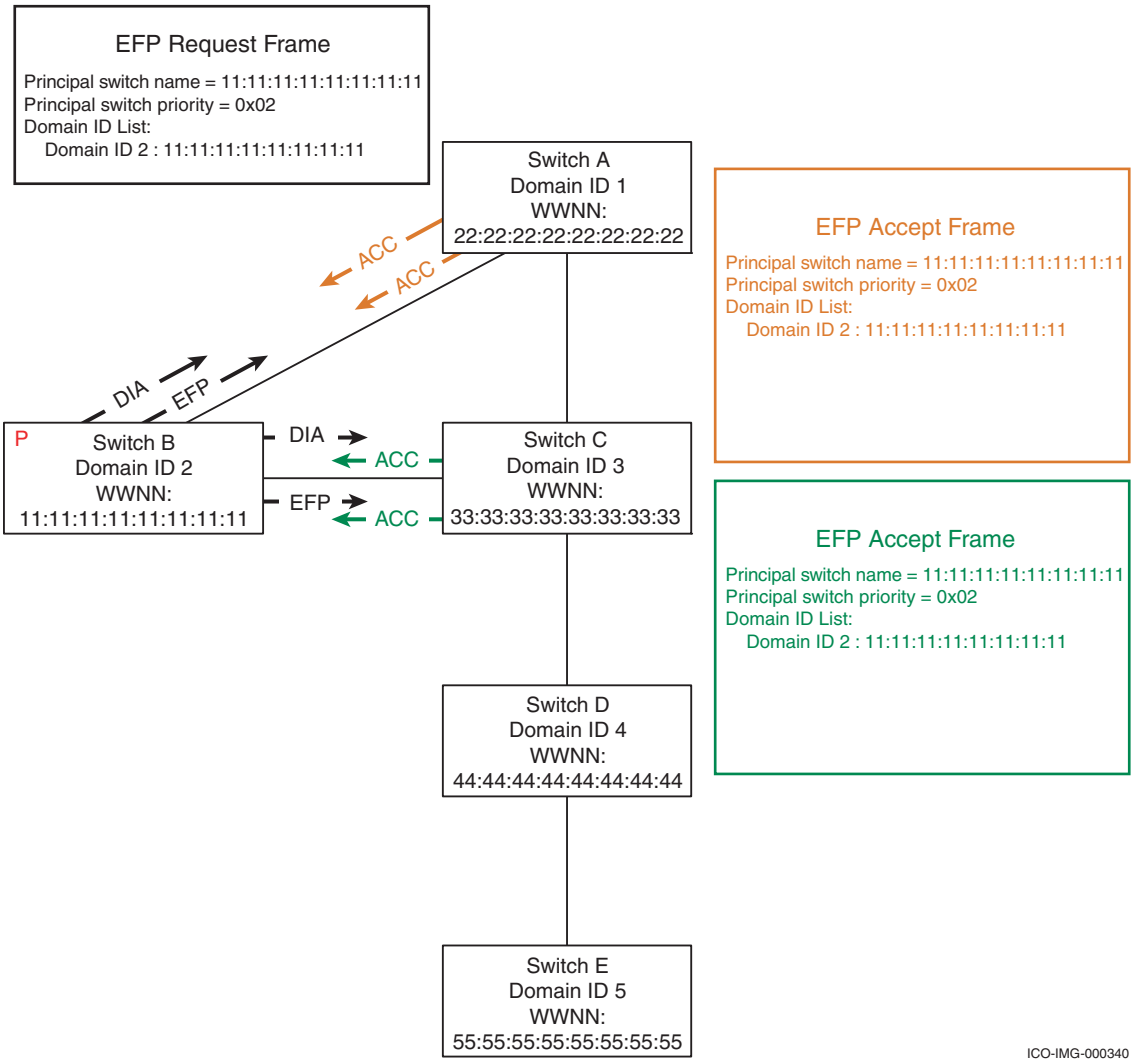


Figure 68 Domain Identifier Assigned (DIA)

RDI – Request Domain Identifier

The RDI allows a switch to request a specific Domain ID or let the principal switch just assign one to it (refer to Figure 69 on page 173). When the principal switch receives the RDI, it takes note of the port on to which it was received to use in the future as the principal downstream ISL. At this point, the principal switch will process the RDI and assign a Domain ID to it, if possible. If a Domain ID is available, an *Accept* will be sent back to the switch that sent the RDI via the principal downstream ISL. Once the Accept has been sent, the principal switch will transmit an EFP containing the updated list of Domain IDs being used. Assuming this is the first RDI to be processed, the Domain ID list in this EFP will contain the principal switches Domain ID and switch name as well as the Domain ID and switch name of the switch that was just assigned the Domain ID. This process will be repeated for each RDI that is processed by the principal switch. The purpose of this EFP is to notify all switches in the fabric of the other Domains and their switch names as they come into the fabric.

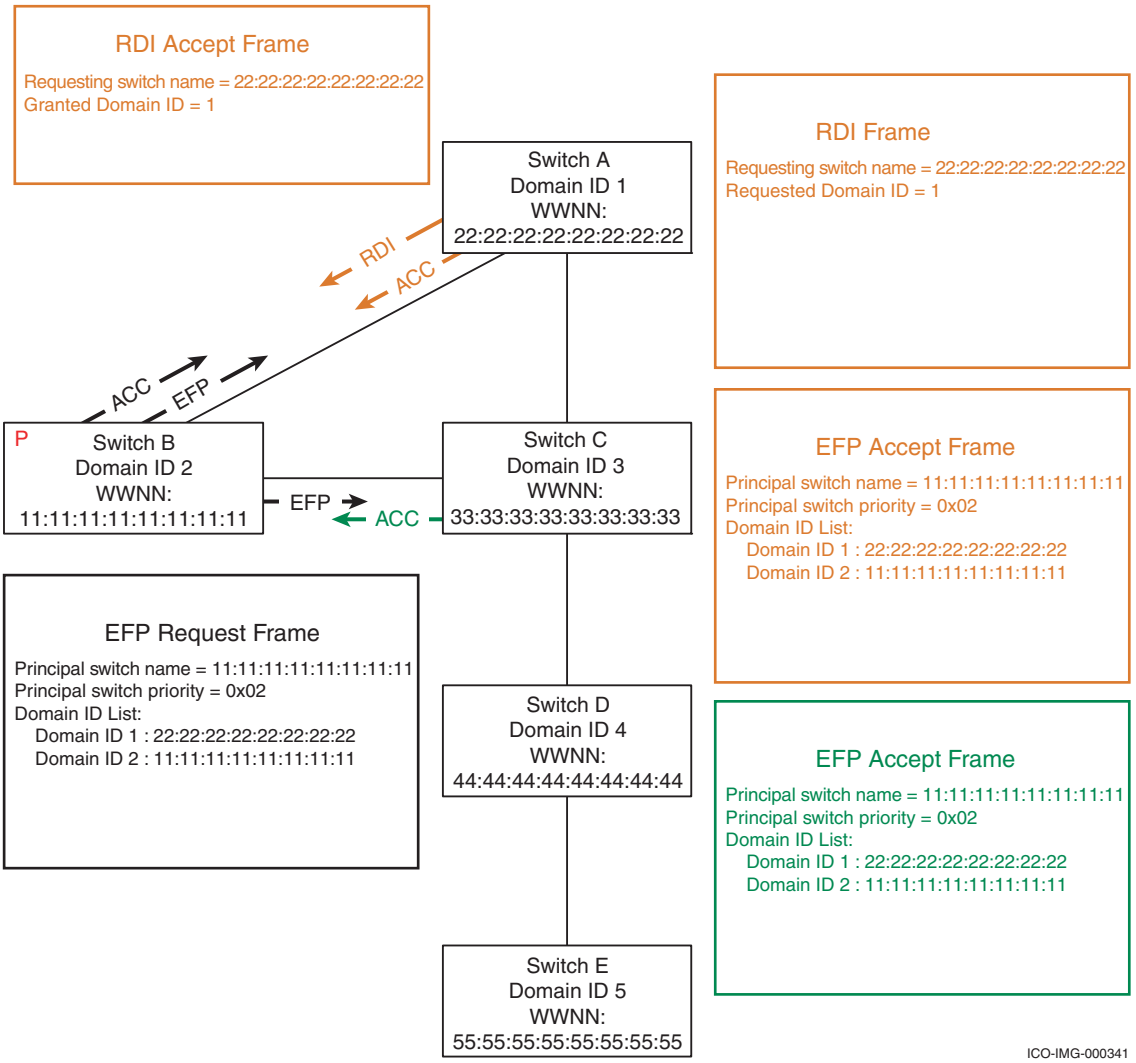


Figure 69 Request Domain Identifier (RDI)

Once the switch has been assigned a Domain ID, it will transmit DIA on all E_Ports other than the principal upstream ISL. These downstream switches will handle the DIA in a manner similar to what is described earlier. However, the RDI from downstream switches will be forwarded to the principal via the principal upstream ISL and the Accept from the principal will be forwarded

back to the originator via the principal downstream ISL. The principal will also generate another EFP to indicate another switch has been assigned an EFP. After the switch receives the Accept to its RDI, it will transmit a DIA on all E_Ports other than the principal upstream ISL.

The process of granting Domain IDs continues until all Domains have been assigned and an additional time period of F_S_TOV expires. The next step in the process is the merge of zoning information.

Zoning merge

After all of the Domain IDs have been assigned, the merge of zoning information is performed. The method in which the merge is completed, and the type of information shared between two switches, varies widely from one platform or operating mode to another. As a result, the information in this section will not be applicable to all environments. However, the diagram from FC-SW-4 and the descriptions listed in Figure 70 on page 175 should give you an idea of how it is performed.

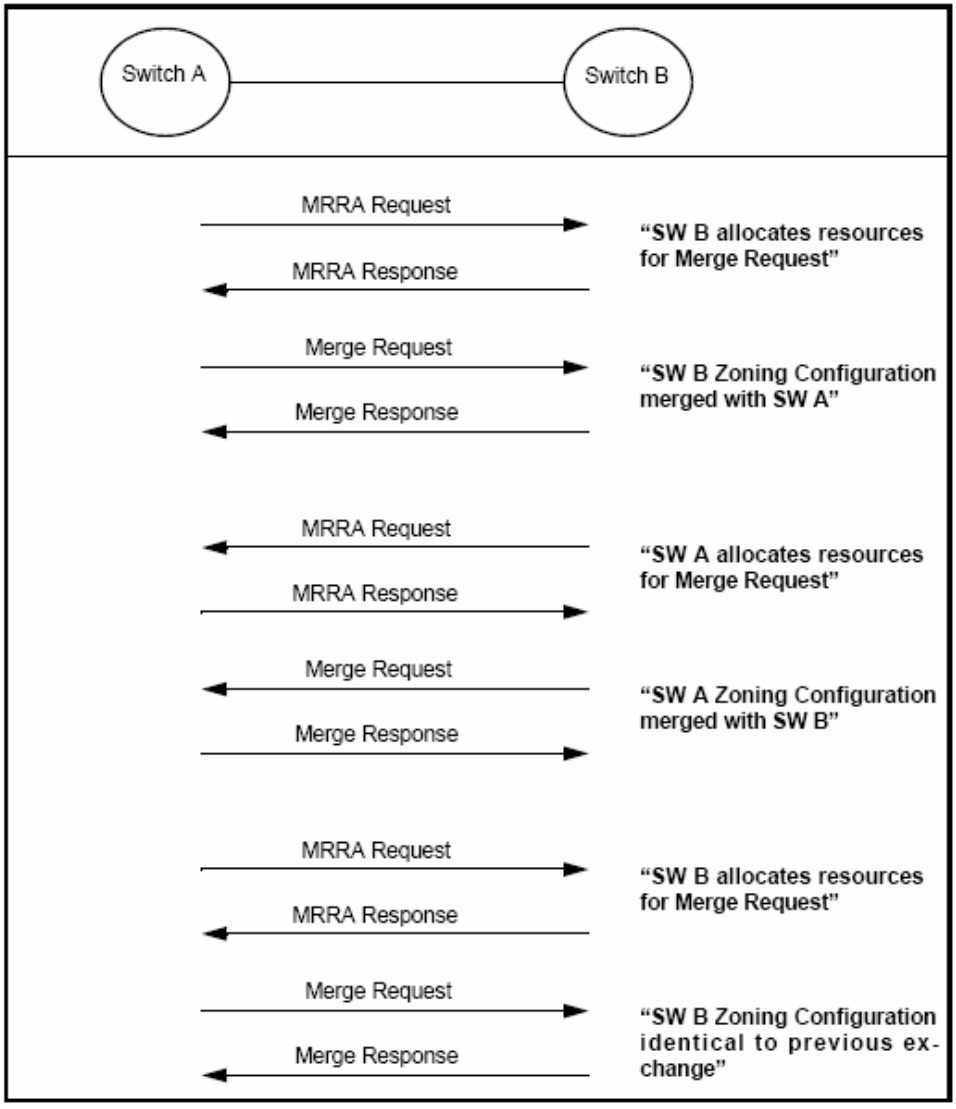


Figure 70 **Exchange of zoning information**

After the ISL has formed between switch A and B and they have both been assigned a Domain ID, the merge process begins with one switch sending another a Merge Request Resource Allocation (MRRA).

MRRA – Merge Request Resource Allocation

The purpose of the MRRA is to ensure that the switch on the other end of the ISL has enough memory available to store the incoming zoning information. When the MRRA is accepted, an Merge Request (MR) is sent from the switch that requested the resource to the switch that accepted the resource request.

MR – Merge Request

The MR contains the actual zoning information, which can consist of differing pieces of information, depending on if enhanced zoning is supported or not. At a minimum, the Merge Request will contain the zone names and zone members for each zone.

Once the zoning information is identical between the two switches, the Fabric Configuration process can continue and path selection begins.

Path selection and the FSPF protocol

When all switches have been assigned a Domain ID and the zoning information has been exchanged, the path selection process can begin. The topology shown in Figure 71 on page 177 will be used for the rest of this example.

Please note that additional ISLs have been added between switch C and D as well as between switch D and E. Also notice that each end of all ISLs have port numbers associated with them. The purpose for these additions is to help facilitate the readers' understanding not only of the path selection process but also to help when we discuss "Trunking and ISL aggregation" on page 209.

Please note that in the next few topology diagrams, the switches icons are represented as different colors.

- **Gray** switches represent switches in an **unknown** FSPF status.
- **Red** switches are **unsettled** from an FSPF algorithm point of view.
- **Green** switches are **settled** from an FSPF algorithm point of view.

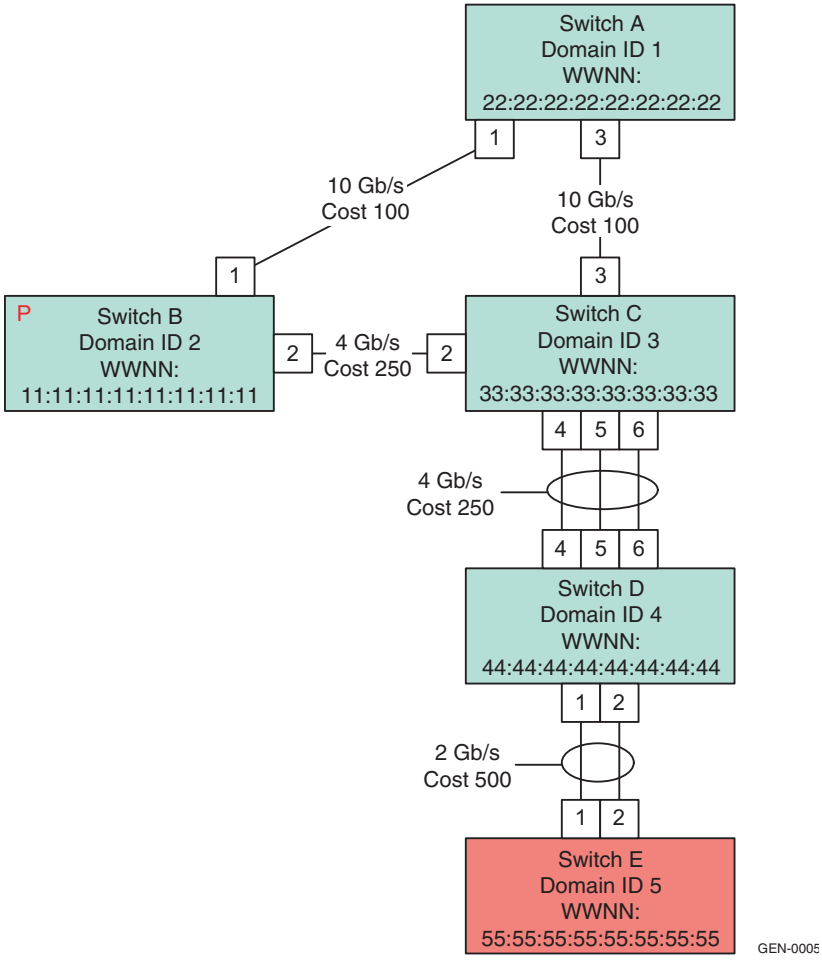


Figure 71 **Path selection topology**

For the sake of simplicity, the path selection process is broken down into five phases, each further discussed in this section:

- "HLO – Hello initial exchange" on page 178
- Database exchange

Note: Refer to "LSU – Link State Update" on page 179 and "LSA – Link State Acknowledgement" on page 181

- "Dijkstra's algorithm for determining the shortest path" on page 182
- "The distributed Name Server, SW_RSCN, and GE_PT" on page 192
- Port assignment

Note: Refer to "Adding Nx_Ports and setting up routes" on page 193.

HLO – Hello initial exchange

Path Selection begins with each switch transmitting HLO out every E_Port. The format of the HLO frame is displayed below on the left and the actual data in the initial HLO from Switch E transmitted out port 1 is shown on the right. In this section, we have tried to avoid showing frame payload contents wherever possible, but the three commands, **HLO**, **LSU**, and **LSA** are used so frequently, and the contents displayed in so many different places in the UIs, that it is advisable to become familiar with how they are laid out.

bits word	31 … 24	23 … 16	15 … 8	7 … 0	
	SOFf				
0	FT-1 Header				
1					
2					
3					
4					
5					
6	Command				FSPF HEADER
7	Version	AR #	Auth Type	Reserved	
8	Originating Domain ID				
9	Authentication				
10					
11	RESERVED				
12	Hello Interval				
13	Dead Interval				
14	Recipient Domian ID				
15	RESERVE	Originating Port Index			
	CRC				
	EOFn				

bits word	31 … 24	23 … 16	15 … 8	7 … 0
	BC	B5	58	58
0	02	FF	FF	FD
1	00	FF	FF	FD
2	22	38	00	00
3	54	00	00	00
4	00	D7	FF	FF
5	00	00	00	00
6	14	00	00	00
7	02	00	00	00
8	00	00	00	05
9	00	00	00	00
10	00	00	00	00
11	00	00	00	00
12	00	00	00	14
13	00	00	00	50
14	FF	FF	FF	FF
15	00	00	00	01
	67	75	4E	D9
	BC	95	D5	D5

Figure 72 Hello frame format and payload

The fields to take note of are:

Command	This is a 4-byte field. For HLO frames it is always set to 14 (00000014).
Originating Domain ID	This is also a 4-byte field and since this HLO is from switch E, it will be set to 05 (00000005).
Hello interval	The length of time in seconds that should elapse between the transmission of each HLO frame. In this case it is set to 0x14 or 20 seconds.
Dead Interval	The length of time in seconds that the switch should wait for an HLO frame before declaring the link dead and removing the link as a valid path from its Link State Database. In this case it is set to 0x50 or 80 seconds.
Recipient Domain ID	Since this is the first HLO being sent between the two switches, neither switch knows the Recipient Domain ID. As a result, it will be set to FFFFFFFF. Once the switch receives a HLO from the other end of the link, it will update this field with the other switches' Domain ID.
Originating Port Index	The port index of the port that transmitted the HLO.

After the first HLOs have been exchanged between all of the switches, they can begin transmitting LSU and perform initial database synchronization.

LSU – Link State Update

The LSU is used to push each switch's Link State Record (LSR) out to the rest of the fabric. There are two basic types of LSUs:

- Database Synchronization — All known LSRs are put into an LSU frame and shared with every other switch that is directly attached.

- Periodic synchronization — Only updated LSRs are put into an LSU frame and shared with every other switch that is directly attached.

In both cases, when a switch receives an LSU from another switch, it will compare the payloads and update its internal Link State Record Database with any LSRs that it either does not have, or are newer than the one it currently has for that Domain. The format of the LSU is shown on the *left* in Figure 73; the data from the payload of the first LSU to be transmitted from switch E is shown to the *right*.

SOFf					BC	B5	58	58
FT-1 Header					02	FF	FF	FD
					00	FF	FF	FD
					22	38	00	00
					4C	00	00	00
					00	E8	FF	FF
					00	00	00	00
Command				FSPF HEADER	15	00	00	00
Version	AR #	Auth Type	Reserved		02	00	00	00
Originating Domain ID					00	00	00	05
Authentication					00	00	00	00
					00	00	00	00
Reserved			Flags		00	00	00	01
Number of Link State Records					00	00	00	01
LSR Type	Reserved	LSR AGE		Link State Record Header # 1 (LSR Type 01)	01	00	00	01
Reserved					00	00	00	00
Link State Identifier					00	00	00	05
Advertising Domain ID					00	00	00	05
Link State Incarnation number					80	00	00	01
Checksum		LSR Length			C4	8D	00	3C
Reserved		Number of Link Desc			00	00	00	02
Link ID				Link Descriptor # 1	00	00	00	04
Reserved	Output Port Index				00	00	00	01
Reserved	Neighbor Port Index				00	00	00	01
Link Type	Reserved	Link Cost			01	00	01	F4
Link ID				Link Descriptor # 2	00	00	00	04
Reserved	Output Port Index				00	00	00	02
Reserved	Neighbor Port Index				00	00	00	02
Link Type	Reserved	Link Cost			01	00	02	F4
CRC					06	50	D2	E1
EOFn					BC	B5	D5	D5

Figure 73 Link State Update frame format and payload

The LSR consists of the Link State Record Header as well as any Link Descriptors associated with the header. In Figure 73, the LSU consists of a single LSR which in turn contains two Link Descriptors. After all of the LSRs have been exchanged, there will eventually be *one* LSR for *each* switch in the fabric (i.e., five switches and five LSRs). Each of the Link Descriptors under the LSR represent one of the ISLs on the

switch that sourced the LSR. For example, a switch with 64 E_Ports would have 64 Link Descriptors in its LSR.

The fields that we need to focus on in order to determine the shortest paths to the other domains are:

Number of link state records	As mentioned previously, there is one LSR for each Domain in the fabric and each Domain is responsible for generating and updating its own LSR. In the previous example, since this is the first LSU being sent out by switch E, there is only one LSR.
Advertising Domain ID	The Domain that is responsible for creating this LSR.
LSR Age	This field is used by each switch to determine if it has the most recent copy of every LSR.
Number of Link Descriptors	The number of active E_Ports on the Domain that created this LSR.
Link ID	The Domain ID of the switch on the other end of the link.
Output Port Index	The port number on the switch that is connected to the other switch.
Neighbor Port Index	The port number on the other switch that is connected to the port on this switch.
Link Cost	The cost of the ISL as determined by: S * (1.0625e12 / Signaling Rate) or 0x03E8 = 1000 = 1 Gb/s = 100 MB/s 0x01F4 = 500 = 2 Gb/s = 200 MB/s 0x00FA = 250 = 4 Gb/s = 400 MB/s 0x0064 = 100 = 10 Gb/s = 1200 MB/s

Once the LSU has been sent out, it will be acknowledged with ACK but not accepted with a normal Accept. Instead, the switch receiving the LSU will send a Link State Acknowledgement (LSA) frame.

LSA – Link State Acknowledgement

When a switch receives an LSU, it will compare the contents with its copy of the Link State Database. If the LSU contains an LSR for a particular domain that is more recent than the LSR it has locally for

that domain, it will replace the existing copy with the new one and forward the new LSR in all subsequent LSUs. Whether or not a newer copy of an LSR is detected, the switch that received the LSU will respond by sending an LSA. Inside of the LSA will be an FSPF header and a copy of each of the Link State Record Headers being acknowledged. Once all of the LSRs have been distributed throughout the fabric, each switch will have a local copy of all LSRs and will be able to determine the shortest path to any destination domain in the fabric.

The algorithm that a switch uses to determine the shortest path is not specified by the FC-SW standard, but it does provide an example of Dijkstra's algorithm, discussed next.

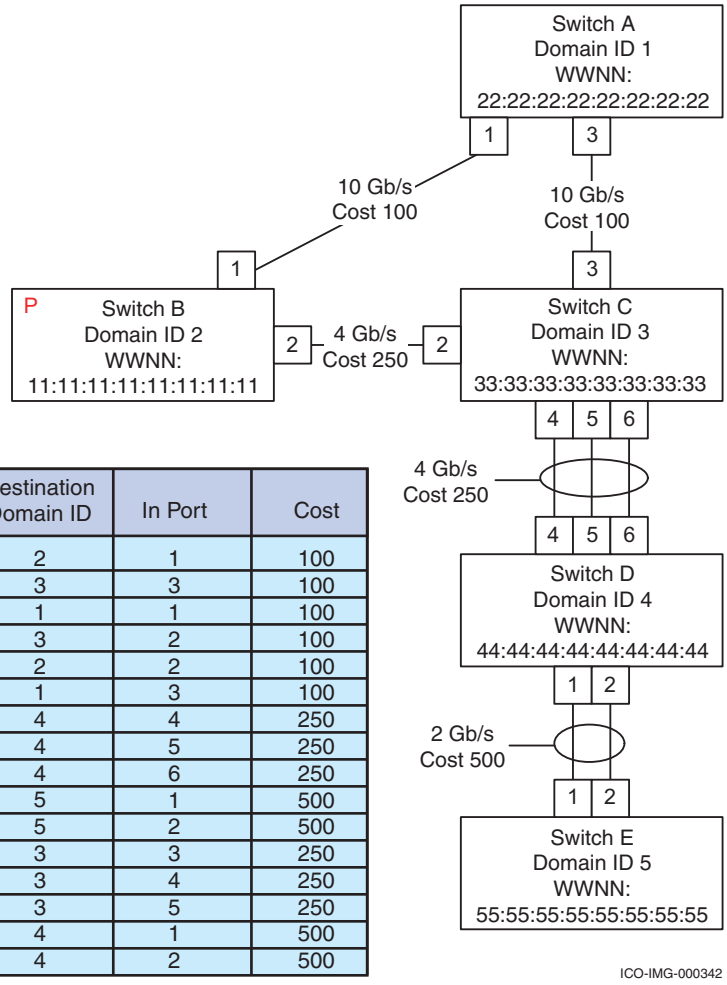
Dijkstra's algorithm for determining the shortest path

Note: The method in which Dijkstra's algorithm presented below is based on an article by Renaud Waldura and can be found at http://renaud.waldura.com/doc/java/dijkstra/

Another good source of information is wikipedia, at http://en.wikipedia.org/wiki/Dijkstra's_algorithm

---

Dijkstra's algorithm allows the user to find the shortest path between two points. In a Fibre Channel fabric, this means the shortest distance between two domains. The result of the LSU and LSA (discussed earlier in this section) will have resulted in a LSR database. For the purposes of our example, a sample LSR database along with the fabric topology in this example is shown in Figure 74 on page 183.



Source Domain ID	Out Port	Destination Domain ID	In Port	Cost
1	1	2	1	100
1	3	3	3	100
2	1	1	1	100
2	2	3	2	100
3	2	2	2	100
3	3	1	3	100
3	4	4	4	250
3	5	4	5	250
3	6	4	6	250
4	1	5	1	500
4	2	5	2	500
4	3	3	3	250
4	4	3	4	250
4	5	3	5	250
5	1	4	1	500
5	2	4	2	500

Figure 74 Sample LSR database and fabric topology

To make this interesting, we will watch the process of Switch B applying Dijkstra's algorithm to the LSR database shown in Figure 74 on page 183. To better understand the process, we will borrow the idea of implementing the algorithm in pseudocode from Renaud Waldura.

The data structures used in the example are as follows:

cost	Stores the best estimate of the lowest cost (shortest path) from the source Domain to each destination domain.
predecessor	Stores the predecessor (previous domain) of each domain on the shortest path from the source.
source	The source domain. In this example, this would be switch B or Domain 2.
S	The set of settled domains, the domains whose shortest distances from the source have been found.
Q	The set of unsettled domains, the domains whose shortest distance from the source have NOT been found.

Start Pseudocode:

```
//initialize cost to infinity
cost = ( 8 )
//initialize predecessor to empty
predecessor = ( )
//initialize S and D to empty
S = Q = ( )

//add source to the set of unsettled domains
add source to Q
//set cost to the source equal to zero since there are no ISLs from the source
to itself
cost(source) = 0

//Loop for as long as there are unsettled domains
while Q is not empty
{
//Get the shortest paths from Q and assign to u
     u = getLowestCostDomain(Q)
//Add the shortest paths to S
add u to S
calculateDistanceToNeighbors(u)
}
```

The functions are listed below:

```
calculateDistanceToNeighbors(u)
{
   //check out each adjacent domain to u as long as the domain isn't settled.
   for each domain d adjacent to u, d not in S
   {
      //if true, a lower cost path (shorter distance) exists
      if cost(d) > ( cost(u) + cost(u-v) )
      {
         cost(d) = ( cost(u) + cost(u-v) )
      }
         predecessor(d) = u
         add d to Q
   }
}

getLowestCostDomain(Q)
{
   find the lowest (in terms of cost) domain in Q
   remove the lowest cost domain from Q
   return the lowest cost domain
}
```

End Pseudocode:

Example: You should follow along with the pseudocode as you read through this example.

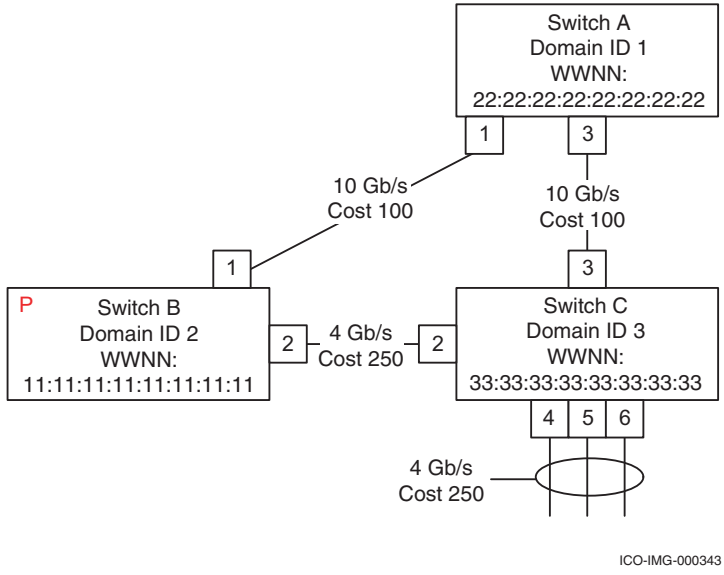


Figure 75 Switch B determining the shortest path

We begin by adding our source switch **B** to the set **Q** (refer to Figure 76). Since Q is not empty, we extract the minimum cost domain B (the only entry in Q, which also has zero cost). We add **B** to **S**, then calculate the distance to its neighbors.

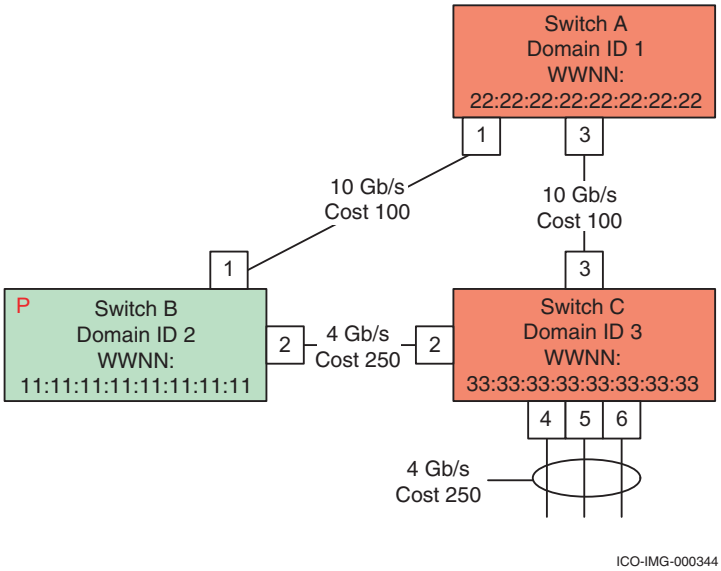


Figure 76 Switch B is settled

As can be seen in the LSR database in Figure 77, those neighbors or domains adjacent to B are **A** and **C** (shown in red in Figure 76 and Figure 77).

Source Domain ID	Out Port	Destination Domain ID	In port	Cost
1	1	2	1	100
1	3	3	3	100
2	1	1	1	100
2	2	3	2	100
3	2	2	2	100
3	3	1	3	100
3	4	4	4	250
3	5	4	5	250
3	6	4	6	250
4	1	5	1	500
4	2	5	2	500
4	3	3	3	250
4	4	3	4	250
4	5	3	5	250
5	1	4	1	500
5	2	4	2	500

Figure 77 **LSR database**

We first compute the best distance estimate from **B** to **A**. Since distance (A) was initialized to infinity, we calculate the distance B to A as follows:

cost(A) = cost(B) + cost(B-A)

cost(A) = 0 + 100

cost(A) = 100

Next, predecessor(A) is set to B and we add A to Q (the set of unsettled domains). Following the same process for C, we assign cost(C) to 250 and predecessor(C) to B.

The second time around, Q contains A and C. As seen in Figure 78, **A** is the domain with the current lowest cost of 100. It is extracted from the queue and added to **S**, the set of settled domains.

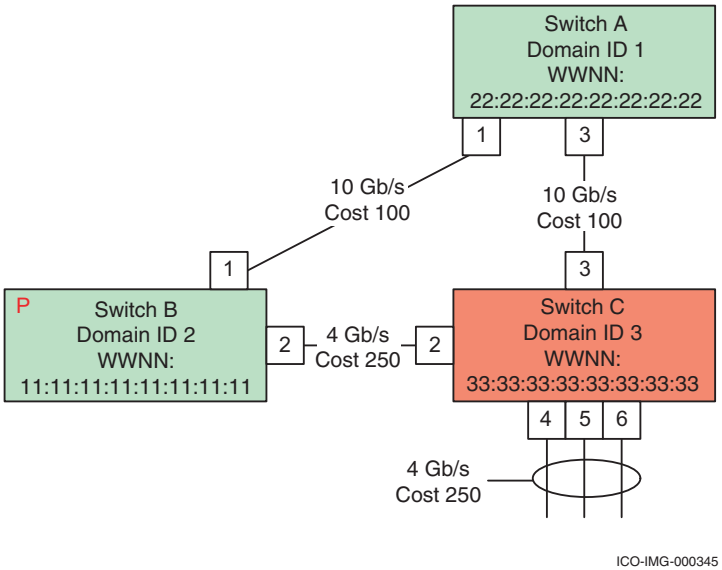


Figure 78 Switch A is settled

Next, we calculate the distance to A's neighbors, **B** and **C**, using the information in the LSR database shown in Figure 79 on page 189.

Source Domain ID	Out Port	Destination Domain ID	In port	Cost
1	1	2	1	100
1	3	3	3	100
2	1	1	1	100
2	2	3	2	100
3	2	2	2	100
3	3	1	3	100
3	4	4	4	250
3	5	4	5	250
3	6	4	6	250
4	1	5	1	500
4	2	5	2	500
4	3	3	3	250
4	4	3	4	250
4	5	3	5	250
5	1	4	1	500
5	2	4	2	500

Figure 79 LSR database

Domain B (Domain ID 2) is ignored because it is found in the settled set. The first pass of the algorithm had determined that the lowest cost path from B to C was direct. However, looking at A's neighbor, **C**, we realize that the following statement evaluates to true and therefore a lower cost path to C must exist:

cost(C) > cost(A) + cost(A-C)

250 > 100 + 100

250 > 200

In fact, we have found that a shorter path going through A exists between B and C. The value of distance(C) is updated to 200 and predecessor(C) is set to **A**. **C** is added again to **Q**.

The unsettled domain with the lowest cost is now **C** and it is extracted from the queue and added to **S** (refer to Figure 80 on page 190).

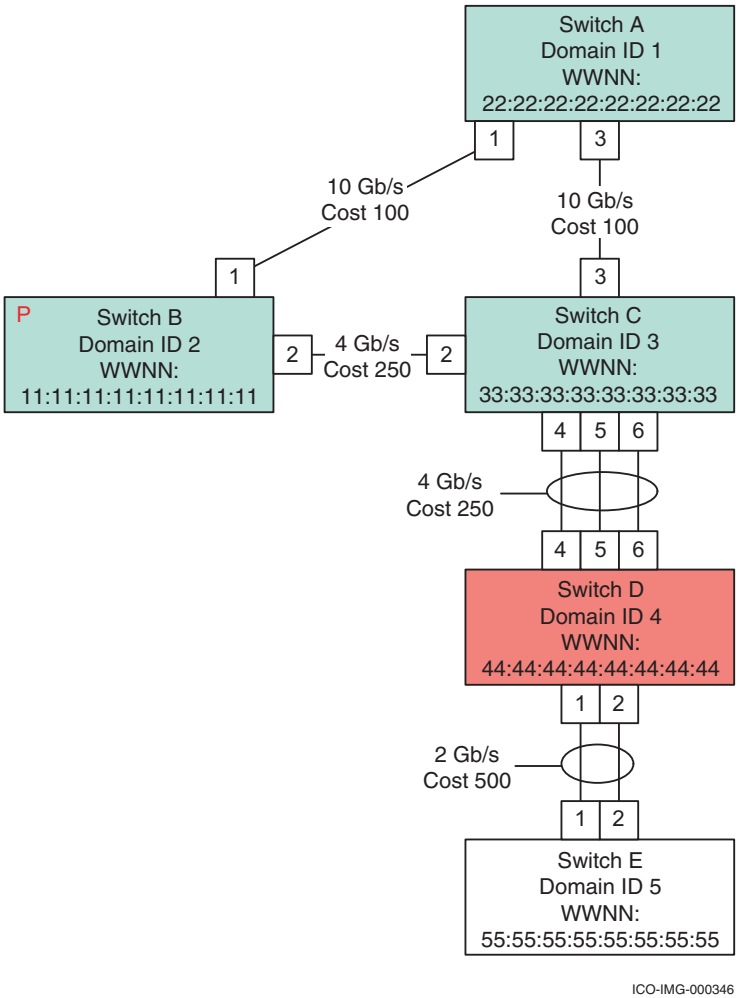


Figure 80 Shortest path

The adjacent switches are determined by referring to the LSR database as shown in Figure 81:

Source Domain ID	Out Port	Destination Domain ID	In port	Cost
1	1	2	1	100
1	3	3	3	100
2	1	1	1	100
2	2	3	2	100
3	2	2	2	100
3	3	1	3	100
3	4	4	4	250
3	5	4	5	250
3	6	4	6	250
4	1	5	1	500
4	2	5	2	500
4	3	3	3	250
4	4	3	4	250
4	5	3	5	250
5	1	4	1	500
5	2	4	2	500

Figure 81 **LSR database**

Domains B (Domain ID 2) and A (Domain ID 1) are ignored because they are found in the settled set. Domain D, however, is not in the settled set so the cost from B to D is evaluated as follows:

cost(D) >	cost(C) + cost(C-D)
250 >	1200 + 250
250 !>	450

Since it evaluates to false, a shorter path does not exist to D. Next, predecessor(D) is set to **C** and **D** is added to **Q**. Finally, the unsettled domain with the lowest cost is **E** and since all of its neighbors are settled, the algorithm ends. Routes can now be determined and the routing tables within the switch can now be programmed.

To view portions of the LSR database on each switch, use the following commands:

- For Brocade FOS — **uRouteShow**
- For Brocade M-EOS — **show fabric topology**

- For Cisco SAN-OS — **show fspf database**

Note that there are the multiple equal cost paths between domains C and D, as well as between D and E. In Fibre Channel, all of these equal cost paths will be used to transfer data. There are multiple ways that this is accomplished, further discussed in "Trunking and ISL aggregation" on page 209. However, the Build Fabric process is not quite complete at this point since the name servers have not yet been synchronized.

The distributed Name Server, SW_RSCN, and GE_PT

In Fibre Channel Fabrics, the Name Server is distributed. This means that every switch in the fabric is responsible for maintaining its own copy of the global name server database. Each switch database is kept in sync in three ways:

- Initial synchronization
- Event driven updates
- Periodic updates:

Initial synchronization takes place when a switch is coming up or joining a fabric for the first time. When this happens, the switch will generate a GE_PT (Get Elements based on Port Type) to every switch in the fabric. It does this by specifying the destination switch in the Destination ID of the GE_PT request frame. For example, if Switch A (Domain ID 1) was sending a GE_PT to Switch B (Domain ID 2), the DID/SID would be FFFC02/FFFC01. In this way each switch can query every other switch in the fabric for its Name Server contents. It is important to realize that when a switch sends a GE_PT to another switch, it has to specify the port type that it is looking for. Typically, a port type of Nx_Port is used, but there have been discovery problems related to the incorrect port type being specified by the requesting switch.

Event driven updates occur when there is a change in the login status of a port. When this happens, the local switch will send a SW_RSCN to each switch in the fabric. It does this by specifying the destination switch in the Destination ID of the switch RSCN (like the GE_PT) and also including in the payload the port ID that changed as well as its current status (offline, online). When the switch receives the RSCN, it may initiate a GE_PT to the switch that sent the SW_RSCN. There have been heterogeneous interoperability issues related to incorrect handling of SW_RSCNs. For an example of this occurring, see the "Adding Nx_Ports and setting up routes" on page 193.

Periodic updates using GE_PT are performed by each switch to ensure that they have the latest copy and did not miss a switch RSCN during the process.

One of the reasons for distributing the name server is so that each switch can service most name server queries locally without needing to initiate a request to another switch in the fabric to satisfy the request. However, on some switch platforms, not all name server database information is shared between all switches in the fabric. An example of this is Symbolic Port Name and Symbolic Node Name information. In some cases when an Nx_Port queries the name server for the Symbolic Port Name or Symbolic Node Name of another port in a different domain, the switch needs to generate a request to the domain in the fabric where that port is logged in. An example of a command that could be used when specific information is needed about a port in another domain is **GA_NXT** (Get All Next).

Adding Nx_Ports and setting up routes

Up until this point, the Build Fabric process has focused on what happens to the switches in the fabric as the Fabric Configuration process is taking place. Now that the Fabric Configuration process has completed, hosts and storage ports can be added and routes can be set up between them.

In Figure 82 on page 194, a single target and initiator are shown. For the sake of this example, we will assume that the target has fully logged in with Switch A and that the initiator has just completed logging in and registered with the name server. Notice that the name server database on both switches contains the entry for target 1 - 10:00:00:00:00:05:00:00 but only the name server on switch B contains both entries. This is because switch B has not transmitted the SW_RSCN yet and subsequently, switch A has not performed the GE_PT.

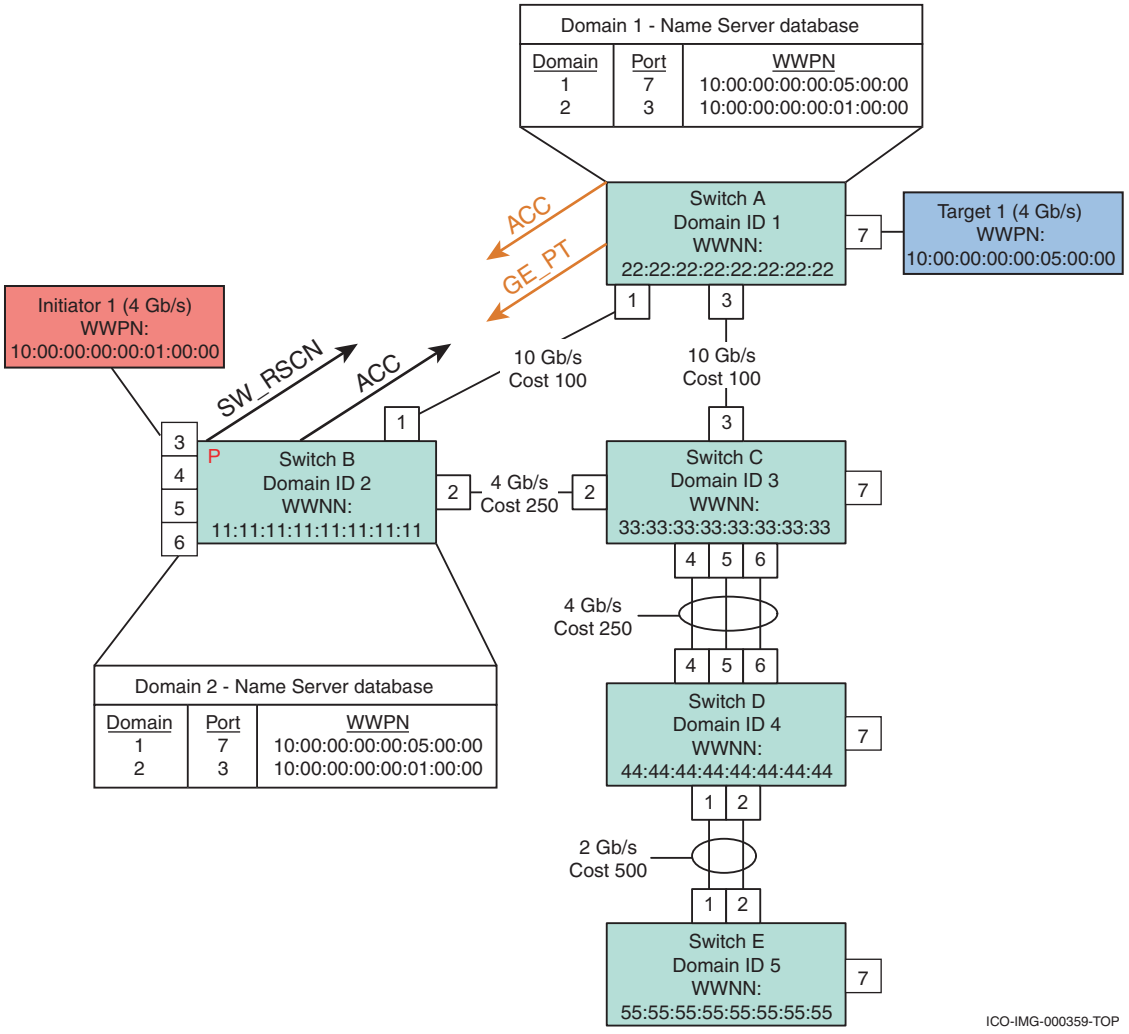


Figure 82 Adding hosts and storage ports and setting up routes

Once the SW_RSCN has been accepted and the GE_PT responded to, both of the name server databases will be in sync. Next, routes can be set up between the two switches.

Assigning egress ports

In "Dijkstra's algorithm for determining the shortest path" on page 182, the LSR database was used to determine the shortest paths. At the end of this process, we were able to determine that the shortest

path off of switch B to any destination in the fabric was via port 1. This would make routing decisions simple since any frame coming into the switch destined for another Domain ID would simply be routed to port 1 and sent off to switch A. If this frame was then received by switch A and was destined to a port on switch A, then it would simply route it to the correct local port. However, if the frame were destined to a domain other than A, and you were to ask what the shortest path off of switch A is, you would have to answer that it depends upon the destination. If the destination were switch B, then the shortest path is via port 1. If the destination were switch C, D or E, then the shortest path would be via port 3.

Determining the shortest path for a frame to take is a decision that each switch must make millions of times per second. One way to make this process as efficient as possible is via the use of Content Addressable Memory (CAM). Regardless of how it is going to be implemented in hardware, some kind of routing table needs to be created that takes into consideration all possible ingress ports (ports where frames are coming into the switch) and egress ports (ports where frames are leaving the switch).

Frequently, switches will have at least a few different routing tables. For example, one routing table would handle frames being received with a destination Domain ID that is different than the local domain ID. The other table may handle frames being received with a Domain ID the same as the local Domain ID. A simple example of a routing table that could be used on switch A in the previous example fabric is shown in Figure 83.

Ingress Port	Destination Domain	Egress Port
1	3	3
	4	3
	5	3
3	1	1
7	1	1
	3	3
	4	3
	5	3

Figure 83 Routing table example

In the simplest case where there is no dynamic load sharing or Exchange-based routing on the ISLs, if there were multiple egress ports to the same domain, some number of ingress ports would be assigned by the switch to use each egress port. This presents a problem since this approach does not take into account the utilization of each ingress port before making the assignment. Because the utilization of each ingress port is not taken into account, there is a very high probability that they are distributed in a manner that does not allow for all of the bandwidth to be used on the egress ports. It should be noted that there are several features in place to help alleviate these problems, all of which will be discussed in the next few sections.

Preferred Paths / Static assignments

As discussed in "Assigning egress ports" on page 194, frequently a switch does not assign ingress ports so that they take full advantage of egress port bandwidth. This is not usually a problem that needs to concern customers, but in some circumstances the unpredictability of the latency through a fabric caused by congestion can cause problems to applications that require large amounts of bandwidth. Because of this, a customer may want to dedicate specific ISLs to service a particular application. This would allow the customer to guarantee isolation and the dedication of bandwidth for the applications that need it.

It could also be that a customer wants to dedicate specific ISLs for EMC replication applications, such as SRDF or MirrorView, or simply differentiate a specific business unit's traffic, such as "human resources" or "'financing" from the rest of the environment. One way to accomplish these goals is to statically assign each ingress port to use a specific egress port and manually configure all or a subset of the routes used by each storage and target.

While it is possible to do this, it is *not* a recommended best practice for several reasons. First, it is a very time-consuming process and very prone to human error. Another concern is failover. If an ISL were to fail, the traffic that was previously isolated on that ISL will be moved to any available ISL. Finally, and most importantly, in the vast majority of cases this is a band-aid approach. In a properly architected SAN, the problems that are trying to be overcome by using this approach will not be an issue in the first place. Rather than try to configure preferred paths to overcome a problem, it would be better to make an architecture change instead.

With these considerations in mind, if you really need to set up static routing, the following information will describe the layout and help you to understand the ramifications and costs of such a solution.

Connectrix B example

General information

On Connectrix B switches, the user has the option to use *port-based routing*. With port-based routing, FSPF is used to determine the shortest path and the switch will define egress ports cost of the link, S_ID and D_ID.

Once the link is established between the S_ID and D_ID, it remains the same throughout the life of the login session. This can introduce the possibility of all high I/O devices being allocated to one ISL since all egress port assignments are not redistributed on every login. Port-based routing does a "best-effort" job of distributing I/O by balancing source port routes but since the switch has no way to predict which routes will be "jammed" when setting up routes, there is a chance of getting multiple high bandwidth ports stacked up on the same ISL. Again, until a switch goes offline, or the fabric changes; all paths will remain the same.

Configuring port-based routing and FSPF

Brocade uses the standard link cost calculations for FSPF with the exception of 2 Gb and 4 Gb port settings in FOS 4.4x. In general the link cost for each setting is:

1 Gb = 1000

2 Gb/4 Gb = 500

10 Gb = 100

4 Gb is typically rated at 250 and is configurable through Brocade FOS, but to keep all things equal, by default 2 Gb/4 Gb were set to 500 to better guarantee the balance of I/O across available ISLs.

It is possible to reconfigure the link speed if one can determine that the cost change will positively impact the flow of traffic in the SAN. The practice of configuring a static route via the port-based policy via the command line interface is shown next:

link cost: [SlotNumber/]PortNumber Cost

From Web Tools you can set the link cost and the route, as well as whether or not frames are set to in-order delivery, as shown in Figure 84.

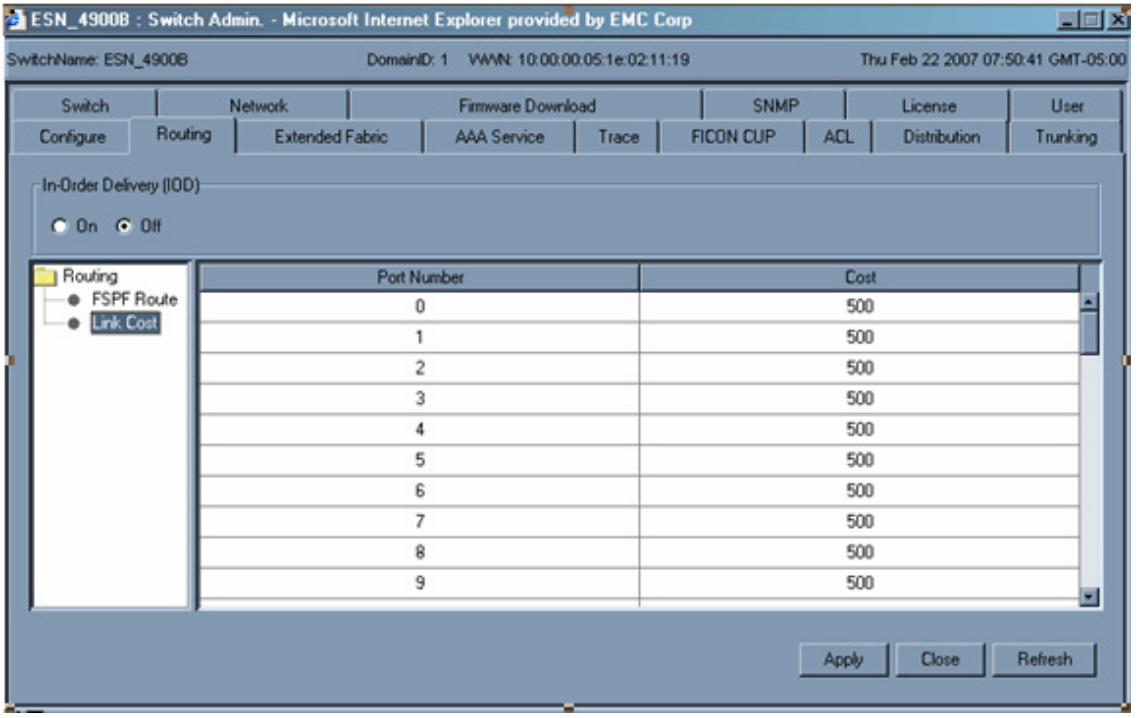


Figure 84 Routing tab

To configure static routes in the switch through CLI:

IMPORTANT

This is a disruptive activity and must be pre-planned.

1. Disable the switch with the **switchDisable** command:

   l82c7219:admin> **switchDisable**

2. Turn on port-based routing using the **aptpolicy** command:

   l82c7219:admin> **aptpolicy**

   Current Policy: 3 [default policy is always exchange-based routing]

3: Default Policy
1: Port Based Routing Policy
3: Exchange Based Routing Policy

l82c7219:admin> **aptpolicy 1**
Policy updated successfully.

3. Configure your static route using the command:

   <urouteconfig> [in port], [destination domain], [out port]

   Example:
   l82c7102:admin> urouteconfig 5 219 0
   l82c7102:admin> urouteconfig 6 219 0

   This has mapped the ingress ports 5 and 6 to destination domain ID 219 via Port 0.

4. To display your routes use the **<urouteshow>** command.

Brocade Fabric OS 5.x and later user manual(s) are good resources for the various settings and configurations that may be used beyond the scope of this document.

Connectrix MDS example

General information

Preferred Path routing in Connectrix MDS was introduced in v3.0(3). Static Routing can be configured through Device Manager. Route control has been integrated in SAN-OS since version 1.x.

Configure Fibre Channel routes

Configure a static route

The first thing that needs to be understood about creating a Fibre Channel route through Device Manager is that FSPF is *disabled*. Static routing in MDS Series fabrics is accomplished through an algorithm that forwards the frame based on the FC ID. By using FC ID for a specific interface and domain, customers can configure a specific route.

To configure a specific route:

1. Log in to Device Manager through **Start> Programs> MDS9000> Device Manager**.
2. From the **Main** menu navigate to **FC> Advanced> Routes**.
3. Click **Create** to add a static route
4. Select the VSAN ID that is going to contain the route.

5. Add the destination address and destination mask for the device being used to configure a route.
6. Select the interface (port) that is going to be used as the egress port to the destination.
7. Select either local or remote options. (Local means the next hop is the final destination; remote means that the next hop is not the final destination.)
8. Click **Create** to save the changes.
9. Click **Close** to abort the operation.

To view all of the Fibre Channel Routes, use Fabric Manager.

Preferred Path routing

The ability to specify routes for various applications can be specified in the Cisco MDS Series switches in the form of preferred path routing. It provides a method to route traffic without the consideration of FSPF. Paths are chosen based on frames received on a selected interface or based on the source FC ID specified. This feature is *not* recommended for best practices, yet it may make sense where logical for isolating applications for business purposes or strategic reasoning.

Preferred paths are defined by route maps and configured on a VSAN basis. The criteria by which routes can be configured are based on S_ID and D_ID and specific to egress traffic only, per switch. This means that a "static" mapped preferred path per switch needs to be configured for traffic to travel from one direction to another. Priorities can be specified per path as well, from 1 to 5 (1 being the highest priority).

IMPORTANT

Setting preferred path is a disruptive action and it is recommended to set the priorities when the switch is offline.

Connectrix M example

General information

The layout for this section is subjective to the display screens throughout the use case. For this section, the general topology is two DS-32M2s and a Symmetrix RDF pair.

Configuring static routing through Preferred Path and Prohibit Dynamic Connectivity Mask (PDCM)

Preferred Path is a standard offering with E/OS 6.01.00. There is no license key requirement.

Preferred Path provides granular route control for data transfer across a multi-switch fabric. The user may create a route between two switches, or over a fabric with multiple ISL hops. Each node can have only **one** Preferred Path between two switches. That is, a node is permitted only one "route controlled exit port." For example, Figure 85 on page 202 displays the route of an HBA in a one hop fabric. The first row under the **Out Port #** column identifies port 20 as the ISL used by this node to get to the next domain.

If there is more than one ISL between switches, the "Out Port" may change if the route tables are recalculated because of a fabric event, such as ISL removal or domain addition. Setting a Preferred Path will maintain a static path even if the switch reboots.

The Preferred Path is configured on a per-switch basis. That is, in a multi-hop fabric, the user must configure the "Out Port" (or exit port) for every switch between the initiator/target pair.

If a node with a Preferred Path fails for whatever reason (e.g., optic failure, ISL disconnect, switch failure), the routes will be recalculated and the node will be given a new path. When the Preferred Path is reestablished, the node will resume usage of that path.

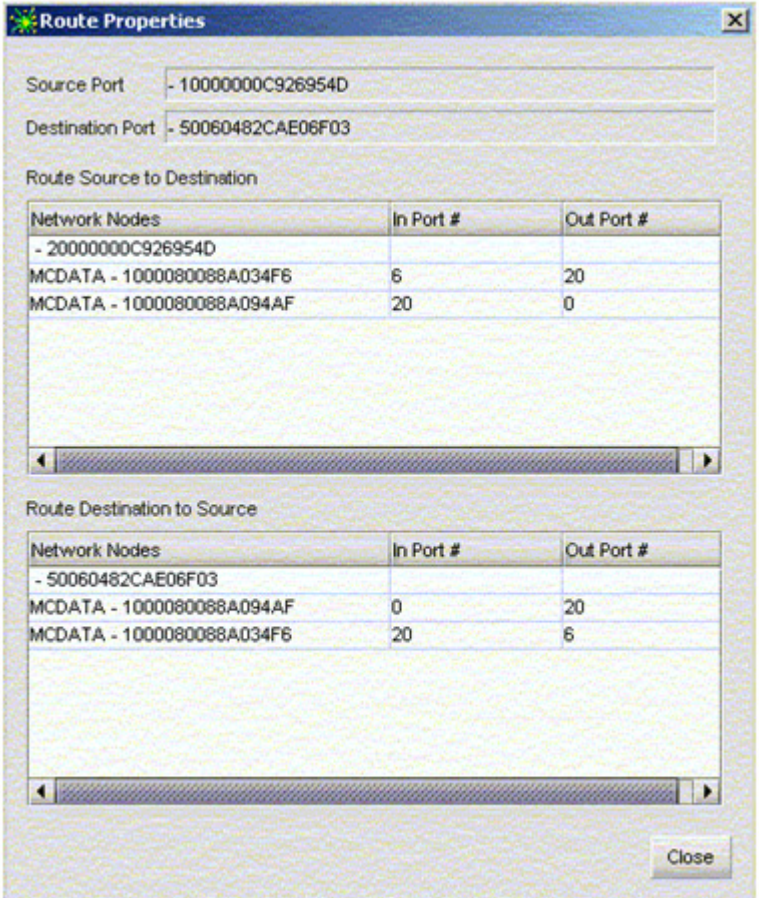


Figure 85 Route properties

IMPORTANT

When planning a Preferred Path, it is important to look at the path from the host and storage perspective. That is, if you want an initiator to have a Preferred Path, you should configure the target to use the same path. This makes for easier management and troubleshooting. It must be configured on both the initiator's switch and the target's switch.

Setting a Preferred Path for one node does not automatically prevent other nodes in the fabric from using the same path. To isolate a node on a Preferred Path, it is necessary to use PDCM (Prohibit Dynamic Connectivity Mask) for E_Ports as well.

Unlike Preferred Path, you may permit or deny a node (initiator or target) from accessing multiple ISLs with PDCM. This provides greater flexibility for the user to restrict the ISLs used by a node. Preferred Path can assign a node to a specific ISL while PDCM will prevent other nodes from accessing that ISL. This has the advantage of dedicating bandwidth to particular nodes.

Configuring Preferred Path and PDCM

Using Connectrix Manager 9.01 or McDATA switch CLI, you can configure a Preferred Path. For CLI commands, please consult the appropriate switch user's guide.

To configure a Preferring Path:

1. Open **Connectrix Manager 9.1** and go to **Start > Program Files > Connectrix Manager 9.1 > Connectrix Manager 9.1**.
2. Log in with the appropriate username and password.
3. Navigate to the desired switch that you want to support Preferred Path and open up **Element Manager** either by double-clicking or by right-clicking **Element Manager**.
4. Using the **Main** menu, choose **Operating Parameters > Switch Parameters > Check Domain ID to "Insistent"**.

   This assures that the Domain ID remains *consistent* through reboots or fabric events and is a *nondisruptive* action.

   Note: ED-10000M configures the Insistent Domain through the **Configure > Domain** tab.

   ---

5. Activate the change and close the GUI screen.
6. Add a Preferred Path.
7. In the Element Manager for the switch, go to **Configure > Preferred Path**.
8. Select the **Enable Preferred Path** box.
9. Click **Add**.

Preferred Path will be configured by port number and not WWN, so the physical port will always be the same unless changed manually in Preferred Path.

10. You have three options:

    - Source Port
    - Exit Port
    - Destination Domain ID

    (Exit port is the E_Port for the ISL to the other domain. D_ID is the next hop. Assure that the correct exit port and Domain ID numbers are used for Preferred Path. See Figure 86.)

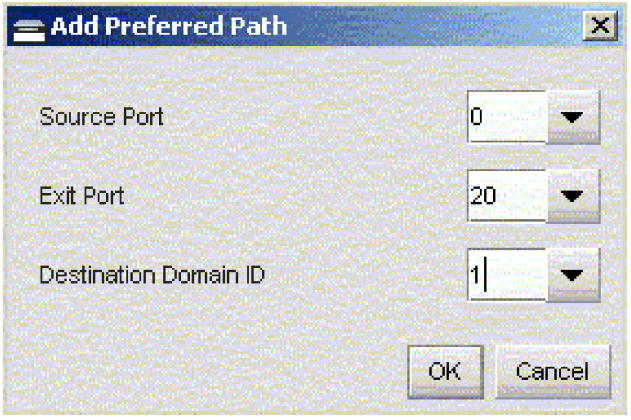


Figure 86 Add Preferred Path

In this example, shown in Figure 87 on page 205, a new route will be created for the Symmetrix SRDF pair between two DS-32M2s. The RDF pair is 6F13 on DS-32M2_bottom and 2ED3 on DS-32M2_top. The columns in the **Product List** window in Connectrix Manager can be manipulated to display the relevant port and domain information. Using the **Show Route** function, the end nodes will be displayed in the **Topology** window. 2ED3 is attached to port 13 on domain 23. 6F13 is attached to port 13 on domain 1.

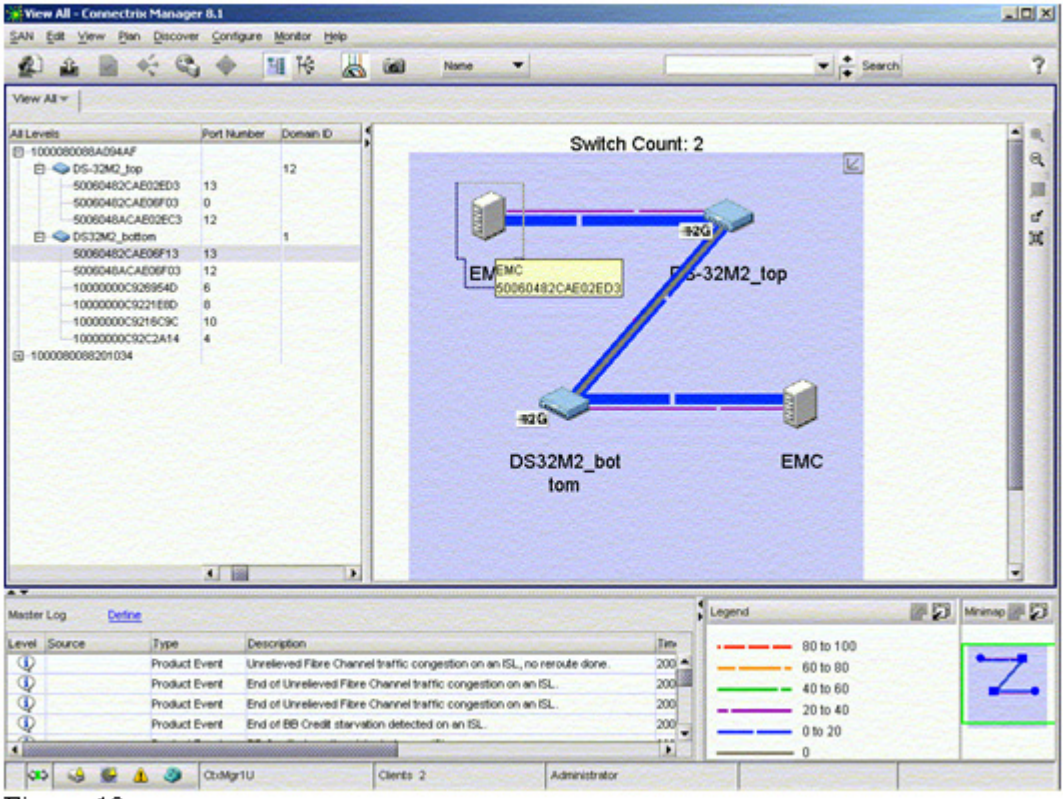


Figure 87 Topology window

The current E_Port used by RDF port 2ED3 can be identified by right-clicking on the link between the node and switch icons. Figure 88 on page 206 shows the route properties before the Preferred Path is configured. It shows that the current path for 2ED3 and 6F13 are using the ISL on port 20 on their respective switches.

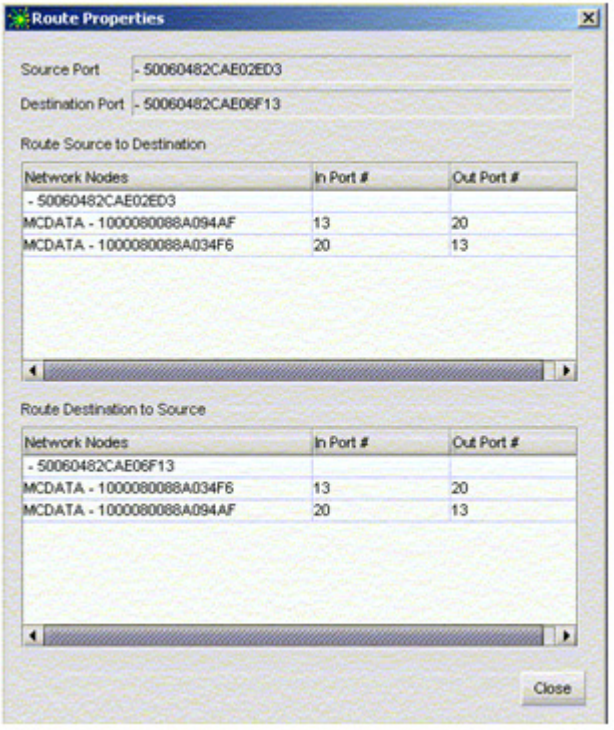


Figure 88 **Route properties dialog box**

This example will change that to the ISL on port 31. Figure 89 on page 207 shows the newly configured Preferred Path windows from DS-32M2_top (top of the screenshot) and from DS-32M2_bottom (bottom of the screenshot). Because of bi-directionality, Preferred Path is configured on both switches for the SRDF pair.

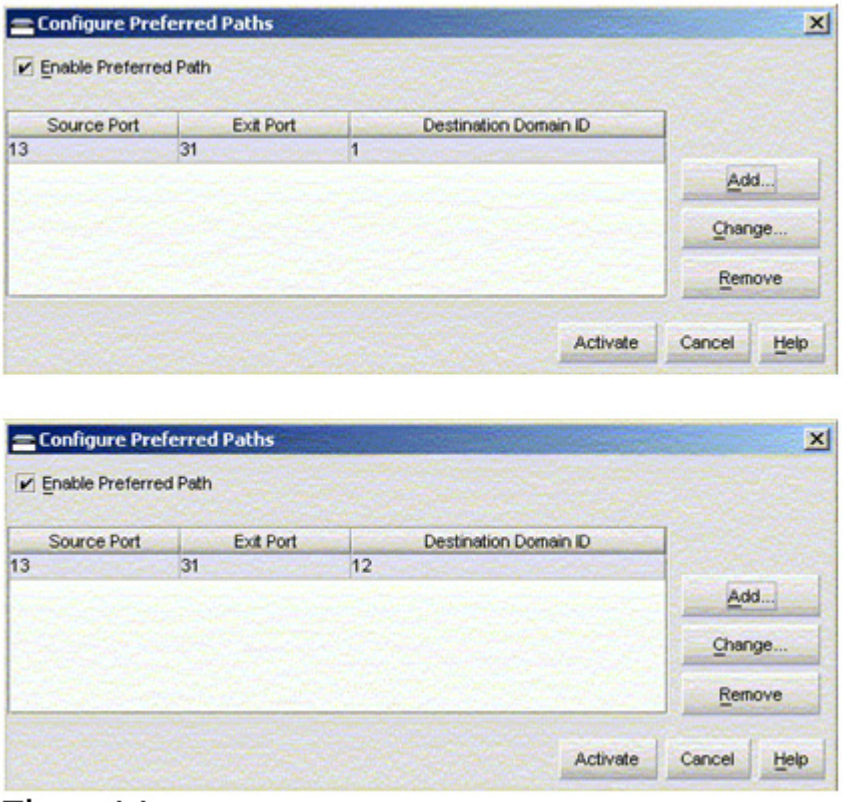


Figure 89 **Newly configured Preferred Path windows**

Trunking

Trunking is a generic term using to describe functionality that allows for more efficient use of ISL bandwidth. "Trunking and ISL aggregation" is discussed on page 209 in this section.

There are three main types of trunking: Flow-based, Frame-based, and Exchange-based, briefly discussed next. These are further developed, including the pros and cons, along with information on other types of trunking, under "Other types of trunking" on page 212.

Main trunking types

Flow-based trunking

This type of trunking was developed by Brocade M Series and is currently marketed as *Open Trunking*. It works by making periodic

changes to the hardware routing tables based on the amount of time being spent at zero BB transmit Credit and a percent utilization measurement. Since the changes are being made to the routing table of the switch transmitting the frames, it can be deployed in any environment and in any interop mode.

Frame-based trunking

Frame-based trunking is available on Brocade B Series (FOS) switches. It involves the aggregation of several physical ISLs between any two adjacent switches into one logical unit so that frames may be evenly distributed across all of the ISLs in a trunk. Once the frames are received on the other side they are reassembled in the proper order and sent to the destination.

Note: For more detailed information on the pros and cons of frame-based trunking, refer to "Frame-based trunking" on page 212.

Exchange-based trunking

Exchange-based trunking is available on Brocade, Cisco, and QLogic switches. It works by distributing exchanges evenly over a group of ISLs. One of the limitations of Fibre Channel is that for the most part, Nx_Ports do not handle getting Out Of Order Frames (OOOFs) within a sequence or exchange very well. Typically, the receiving port will abort the exchange and force the entire exchange to be re-driven. This is unfortunate for trunking implementations, because a trunking solution that could evenly transmit the same number of bytes over each ISL would be very efficient.

The next best thing would be one that could transmit the same number of frames over each ISL. The downside to doing this is that special hardware is needed on the receive side to ensure that the frames are reassembled into the proper order (to prevent OOOFs). By implementing trunking at the exchange level, exchange-based trunking prevents frames within an exchange from getting out of order by routing all of the frames for the exchange down the same ISL.

Note: For more detailed information on the pros and cons of Exchange-based trunking, refer to "Exchange-based trunking" on page 214.

Trunking and ISL aggregation

As previously discussed, the FSPF protocol is used to find the shortest path between two domains. Once this is done, a routing table can be created so that a frame received by an FC switch can be forwarded on to the next hop on its way to its destination.

When Fibre Channel switches were first released, the process stopped here, and similar to Brocade's port based routing when Dynamic Load Sharing is disabled, as discussed in the "Connectrix B example" on page 197, once an assignment was made the route was never moved unless something major in the fabric changed. The main problem with this approach is that as new ports are added, they are assigned to an egress port in a round-robin fashion. This means that as ports are added and removed, the ISLs can become imbalanced. For example, in the configuration shown in Figure 90, there are four initiators and four targets. Each initiator on switch B is zoned to see one of the targets on switch A.

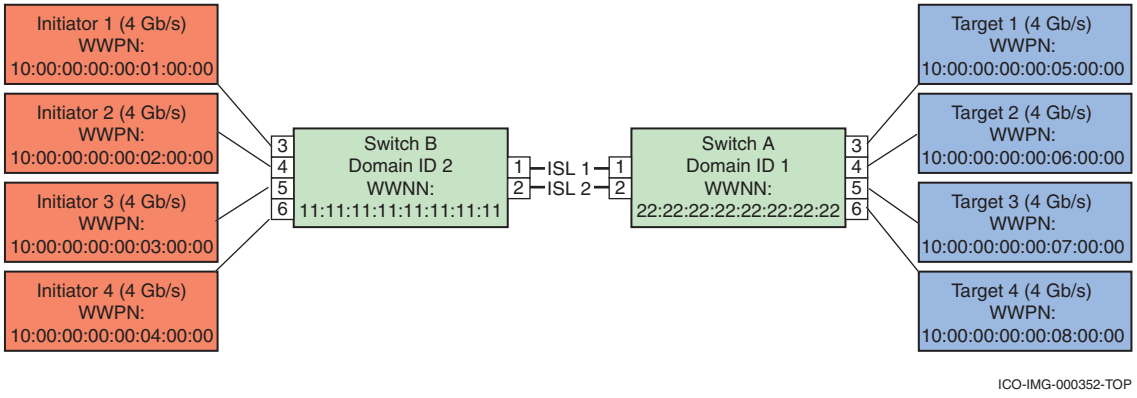


Figure 90 Configuration example

When the initiators and targets come up for the first time, they will be assigned to egress ports in a round-robin fashion, meaning that the first port will be assigned to ISL 1, the second port to log in will be assigned to ISL 2, the third port to log in will be assigned to ISL 1, and the final port will be assigned to ISL 2. See Figure 91 on page 210 where the arrows indicate which ISL each port is assigned to.

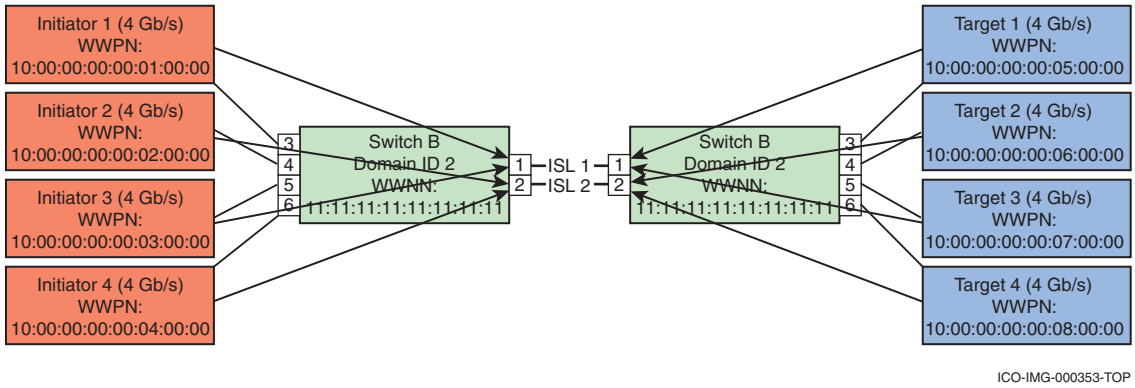


Figure 91 Port assignment example

In Figure 91, the configuration is completely balanced, but what would happen if one of the initiators were to be rebooted? The answer is that after the reboot, as the initiator was coming up, it would be assigned an exit port of ISL 1 since the last assignment was to ISL 2.

The configuration after initiator 2 is rebooted as displayed in Figure 92.

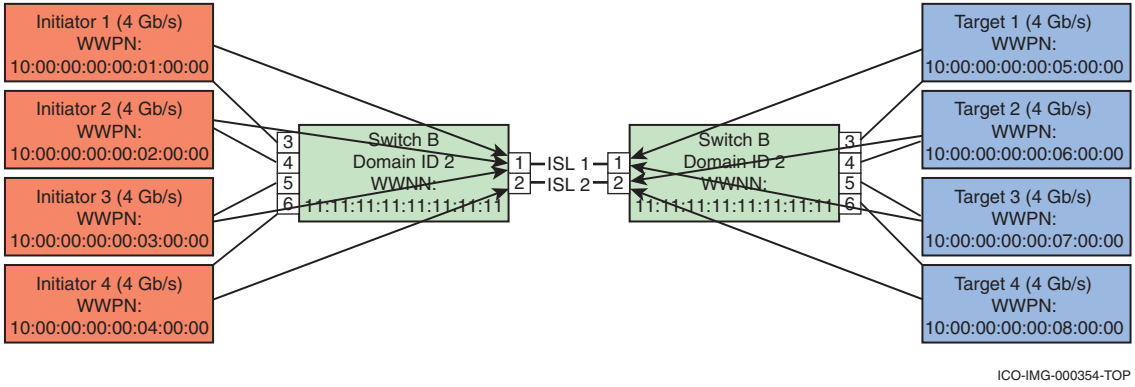


Figure 92 Configuration after initiator 2 is rebooted

One can easily see how this sort of approach could rapidly lead to imbalances in ISL utilization. As a result, this type of functionality did not last for very long, at least not unless a dynamic rebalancing feature is implemented, like Brocade's Dynamic Load Sharing (refer to "Dynamic Load Sharing (DLS)" on page 211 for more information

on DLS).

Round-Robin login with dynamic reassignment

The next major improvement in terms of assigning egress ports was to have the egress ports recalculated whenever a port logged out of a switch. This dynamic reassignment is done in such a way as to prevent Out of Order Frames (OOFs) from occurring. This approach also takes into consideration how many ports are already assigned to each ISL.

This feature is in use on Brocade M-EOS products today and is not configurable. On Brocade FOS products this feature is known as Dynamic Load Sharing (DLS).

Dynamic Load Sharing (DLS)

As mentioned previously, egress port assignment is generally based on the incoming port and the destination domain. This means that all the traffic coming in from a port (either an E_Port or an Fx_Port) directed to the same remote domain is routed through the same output E_Port.

If DLS is turned **off** (using **dlsReset**), load sharing is performed only at boot time or when an Fx_Port comes up. By disabling DLS, the possibility of dropped frames is eliminated every time a change in the fabric occurs. A change in the fabric is defined as an E_Port going up or down, or an Fx_Port going up or down.

If DLS is turned **on** (using **dlsSet**), when there are multiple equivalent paths to a remote switch traffic is shared among all the paths. Load sharing is recomputed when a switch is booted up or every time a change in the fabric occurs. With DLS enabled, traffic on existing ISL ports might be affected when one or more new ISLs are added between the same two switches. Specifically, adding the new ISL might result in dropped frames as routes are adjusted to take advantage of the bandwidth provided by the new ISL.

Enabling DLS optimizes fabric routing. For example, if an Fx_Port goes down, another Fx_Port might be rerouted from one E_Port to a different E_Port. The switch minimizes the number of routing changes, but some are necessary to achieve optimal load sharing. This is the factory default on all Brocade branded switches.

To view and change this parameter.

1. Enter the **dlsshow** command to view the current DLS setting.

   One of the following messages appears:

- *DLS is set.*

  This message means that the DLS option is turned on. Load sharing is reconfigured with every change in the fabric.

- *DLS is not set.*

  This message means that the DLS option is turned off. Load sharing is only reconfigured when the switch is rebooted or an Fx_Port comes up.

2. Enter the **dlsset** command to *enable* Dynamic Load Sharing.
3. Enter the **dlsReset** command to *disable* Dynamic Load Sharing.

It is important to note that by default, this approach does not take into consideration how much bandwidth each ingress port is consuming. However, add-on functionality is available to enable it to look not only at utilization, but also the amount of time spent at zero BB_Credit.

Other types of trunking

Frame-based trunking

Soon after dynamic reassignment had been introduced into FC Fabrics, Brocade released its first version of trunking at the frame level. The approach requires that some of the ISLs between two domains be grouped onto the same quad of ports and then as frames come in assign them to an output port in round-robin fashion. The frames are then reassembled into the proper order on the other side of the ISLs. This approach ensures that the maximum ISL utilization is achieved by spreading the load out evenly across all ISL. Recently, frame trunking was enhanced between platforms that both support 4 Gb/s. Instead of distributing frames evenly over the ISLs in the trunk, it fills a link and then moves to the next link. While frames are not evenly distributed across all ISLs in a 4 Gb based frame trunk, the end result is the same — congestion of the frame trunk does *not* occur until *all* of the ISL's bandwidth is used. Refer to Figure 93 and Figure 94 on page 213 for an example before and after frame-based trunking.

Note: Due to HA considerations, it is better to have two smaller trunks between two domains than one large trunk. For example, if your environment requires 1600 MB/s of bandwidth between two domains, you should configure two 800 MB/s trunks instead of a larger 1600 MB/s trunk.

Another aspect of frame based trunking is that from an FSPF point of view; there only appears to be a single ISL between the two switches per frame trunk.

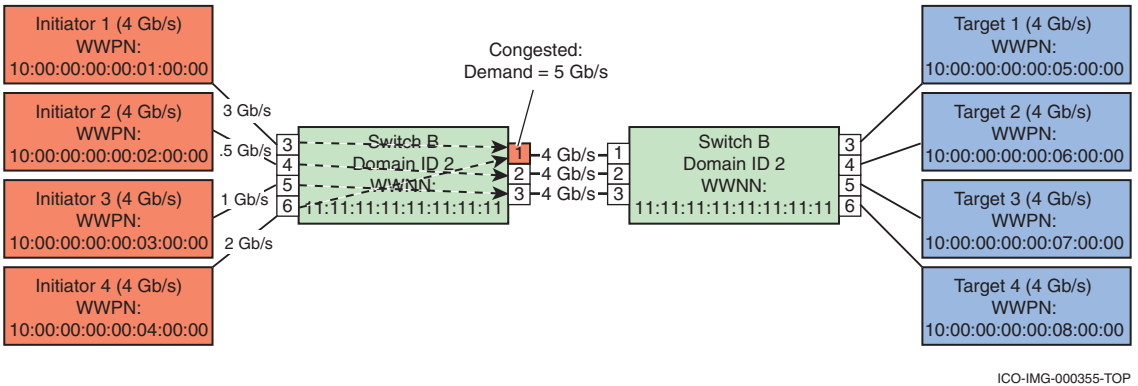


Figure 93 **Before frame-based trunking**

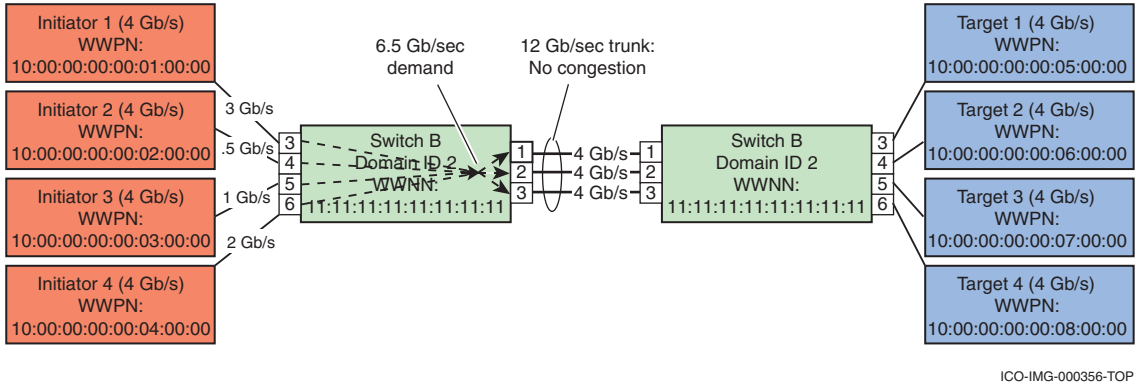


Figure 94 **After frame-based trunking**

Observe the following criteria for standard distance trunking:

- There must be a direct connection between participating switches.
- Trunk ports must reside in the same port group.
- Trunk ports must run at the same speed (either 2 Gb/s or 4 Gb/s).
- Trunk ports must be set to the same ISL mode (L0 is the default).
- Trunk ports must be E_Ports or EX_Ports.

- For optimal performance, cable lengths of no more than 30 meters difference are recommended.
- The switch.interopMode parameter must be set to 0.
- The port ISL mode must be disabled (using the **portCfgIslMode** command).

Note: For more information on trunking, refer to the *Brocade FOS 5.3 Administrators Guide.*

Through the use of special hardware to assist with the re-assembly of frames into the proper order, fame level trunking does not increase the likelihood of receiving OOFs.

PROs:

- Maximum potential utilization of ISLs
- Easy to configure

CONs:

- All ports need to be located on the same port group in order for a trunk to form
- Only works in Brocade Native environments

Exchange-based trunking

The next best thing would be one that could transmit the same number of frames over each ISL. The downside to doing this is that special hardware is needed on the receive side to ensure that the frames are reassembled into the proper order (to prevent OOOFs). By implementing trunking at the exchange level, exchange-based trunking prevents frames within an exchange from getting out of order by routing all of the frames for the exchange down the same ISL without requiring hardware-assisted exchange reassembly.

When Cisco first released its first FC switch, it came with a new kind of trunking that distributed I/O across ISLs based on the Source ID, Destination ID, and Exchange ID. Since its introduction, this method has been adopted by QLogic as well. The concept is very similar to Brocade's frame level trunking but instead of distributing the *frames* equally over all ISLs, *exchanges* get distributed over all ISLs. (Exchanges consist of sequences; sequences consist of frames.)

PROs:

- Works on heterogeneous ISLs (different vendors on each end)
- Does not require that each port be located on the same port group

CONs:

- Downstream congestion can potentially impact all members of the trunk.

IMPORTANT

This type of trunking only balances traffic on egress ports. For this reason, if one of the switches does not support a trunking functionality of some kind, the load will still be optimized, but only in one direction.

Port Channels

Another feature that Cisco released with their first director is the Port Channel. While a Port Channel is not actually a form of trunking (since exchange-based trunking can be done with or without a Port Channel being present), it is important to mention. This is a form of aggregating ISLs so they appear to be a single high-bandwidth ISL from the perspective of FSPF.

PROs:

- Ease of use — Logically grouping all ISLs between two domains enhances the users' ability to administer all of them simultaneously.

CONs:

- Slightly more complicated to configure than normal ISLs. If multiple Port Channels exist between two domains, the user needs to ensure that the FSPF cost is the same or else the higher-cost Port Channels will not be used.
- Since the default cost of a port channel is determined by <Link speed>/number of ISLs, the user needs to be mindful of how the introduction of a low-cost path will impact their topology.
- If a single ISL in a Port Channel goes offline, all frames queued up to use that Port Channel will be discarded. These events are typically very rare and the Upper Layer Protocols will be able to re-drive the I/O that was impacted.

Brocade DPS — Enhanced Exchange-based trunking

DPS is similar to the Exchange-based trunking done by Cisco and QLogic with the exception that instead of making routing decisions based only on the hash of the D_ID, S_ID and OXID, another parameter, RxPort, is used. This is significant because by adding a

fourth piece of data to the hash, the chance of encountering collisions is reduced, theoretically increasing the efficiency of the feature.

PROs:

- Can work on top of ISLs and/or frame trunks.
- Works on heterogeneous ISLs (different vendors on each end)
- Does not require that each port be located on the same port group

CONs:

- Downstream congestion can potentially impact all members of the trunk.

Downstream congestion can potentially impact all members of the trunk.

IMPORTANT

This type of trunking only balances traffic on egress ports. For this reason, if one of the switches does not support a trunking functionality of some kind, the load will still be optimized, but only in one direction.

Open trunking

The last type of trunking to be introduced into FC Fabrics was McDATA's Open trunking. Open trunking works by making periodic updates to the switch routing table. The updates are made based on the percentage of ISL utilization, the amount of time spent at zero BB_Credit, and the source port's utilization percentage. Open trunking attempts to find the best mix on each ISL to minimize congestion. Because updates are made to the switch routing table, an entire ingress port to a given domain (also known as a flow) is moved instead of one exchange or frame. On the Brocade ED-10000, the feature was improved to not only consider the ingress port and destination ID. but also the port on the destination ID.

PROs:

- Works on heterogeneous ISLs (different vendors on each end)
- Does not require that each port be located on the same port group
- All ports do not need to be located on the same port group in order for trunking to work.
- Downstream congestion will not impact all members of the trunk. See "Congestion and backpressure" on page 217.

CONs:

- The periodic changes to the switch routing table will occasionally result in a small number of OOFs.
- Open Trunking not capable of fully saturating all ISLs since the level of granularity only goes down to the flow level and not to the exchange or frame.

IMPORTANT

This type of trunking only balances traffic on egress ports. For this reason, if one of the switches does not support a trunking functionality of some kind, the load will still be optimized, but only in one direction.

Congestion and backpressure

The terms *congestion* and *backpressure* are sometimes used interchangeably, but although they are closely related, they are very different.

- *Congestion* occurs at the point of restriction.
- *Backpressure* is the effect on the environment leading up to the point of restriction.

To help explain the concepts of congestion and backpressure, we will use a mass transit system. For the sake of this backpressure example, assume that a bus leaves from terminal "A" destined for terminal "B" once per minute and each bus is capable of carrying 40 passengers at a time (Figure 95 on page 218). As long as the number of passengers arriving at terminal "A" does not exceed 40 passengers a minute, the system will work perfectly fine. The busses will carry less than their full load of 40 passengers per minute and there will be no passengers waiting in the queue.

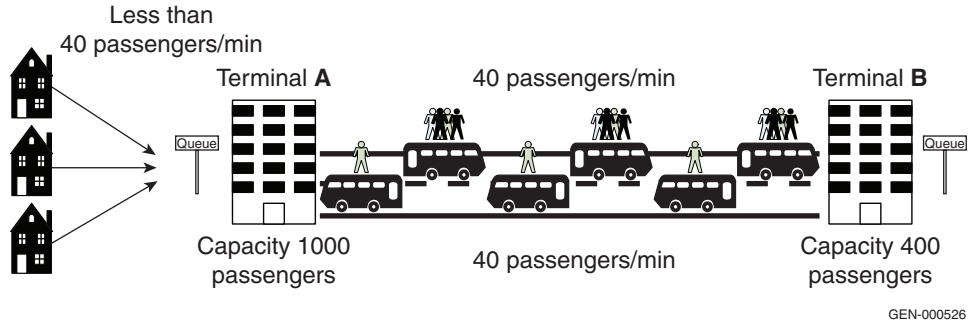


Figure 95 **Backpressure example using mass transit system**

However, when the number of people arriving at terminal "A" exceeds 40 passengers a minute, (even by a small number, like 42 passengers a minute), a number of things happen (see Figure 96).

- The first impact on the environment is an increase in latency through the system. In other words, it will take more time for a passenger to get from terminal "A" to terminal "B" since each passenger will have to wait in a queue for some period of time.
- Occurring in parallel with this increase in latency is an increase in the amount of "buffer" space being consumed by the number of passengers waiting for a bus. Since each passenger consumes some amount of "buffer" space, and each passenger is waiting longer due to the increased latency, the amount of buffer space used increases over time.

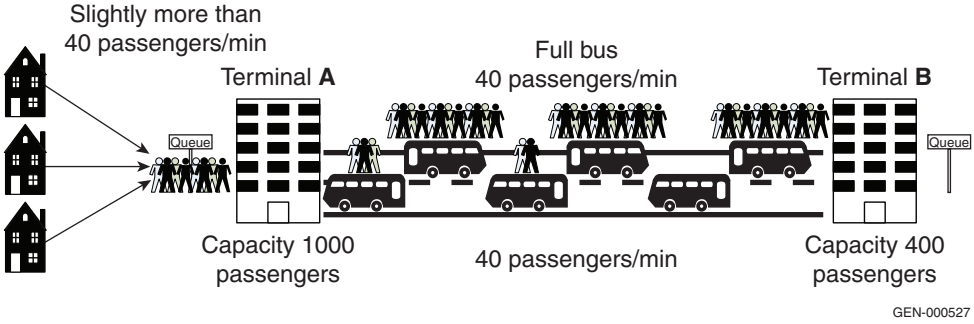


Figure 96 **Increase in consumption of buffer**

As shown in Figure 97, if passengers continue to be arrive at a rate faster than 40 per minute, the "buffer" would eventually fill up and overflow (people would have to wait outside the terminal).

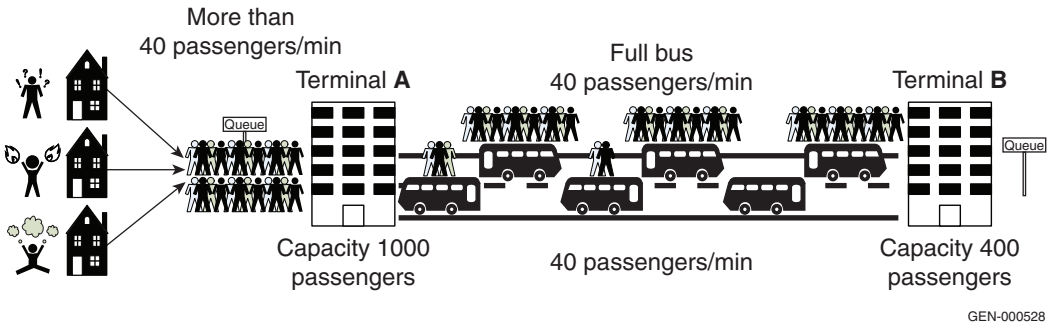


Figure 97 Buffer filled, causing overflow

In the case of a Fibre Channel link, because of the use of buffer-to-buffer flow control the buffer does not overflow. Instead, a phenomenon known as *backpressure* is experienced. Think of an terminal employee notifying people planning to come to the terminal that there is no space available and you have an idea of how flow control works.

For the rest of this section, the cause and effects of congestion and backpressure are investigated as well as how to judge if your environment is inherently susceptible to these issues. In addition, information on how to detect these conditions is provided.

To make things easier to explain, the topology shown in Figure 98 is used.

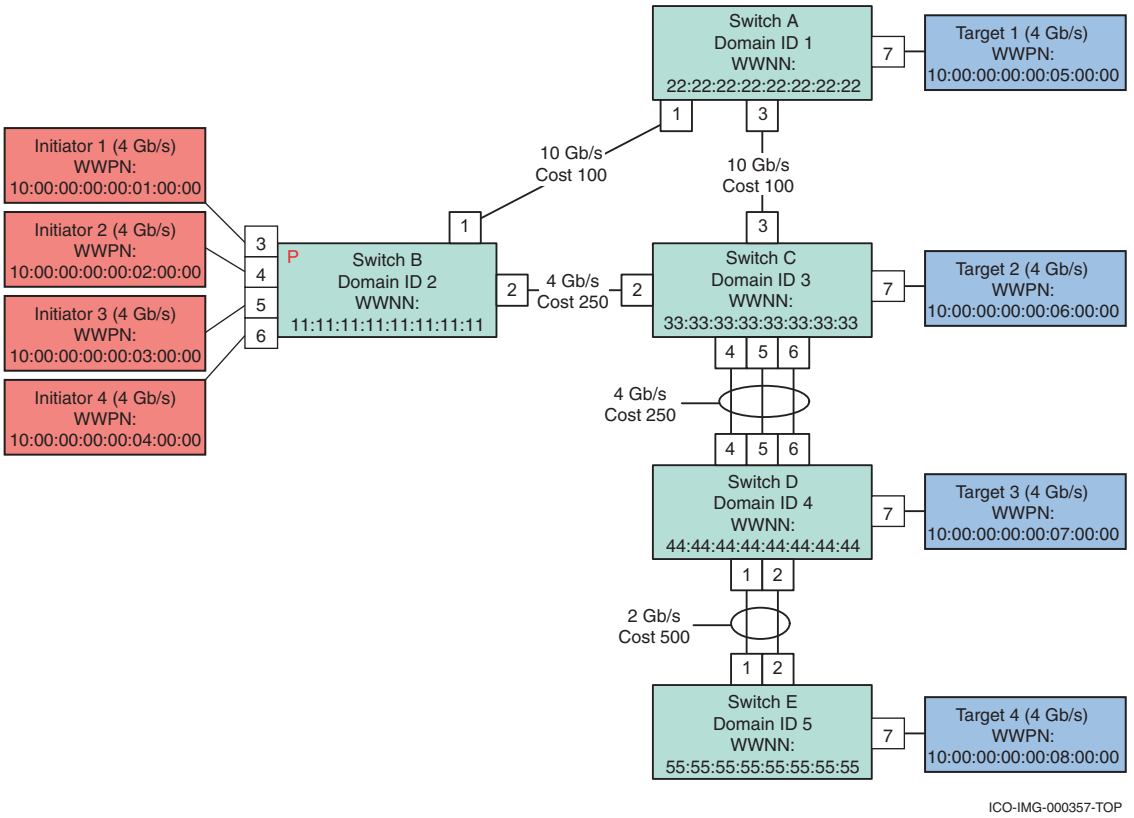


Figure 98 Topology example

For the rest of this example, assume that each initiator is zoned so that it only has access to one target; for example:

- Initiator 1 — Target 1
- Initiator 2 — Target 2
- Initiator 3 — Target 3
- Initiator 4 — Target 4

If we take this configuration and apply the same sort of situation to it as we did in the terminal example, you will gain a better understanding of how flow control in an FC fabric works. Before we

do this, we will first drill down into the case where only Initiator 1 is sending frames to Target 1 at 4 Gb/s.

Buffer-to-buffer flow control

Buffer-to-buffer flow control is the mechanism used to ensure that a transmitter does not overwhelm the receiver with too many frames. It relies on the use of credits that are exchanged at login time (FLOGI for Nx_Ports, ELP for E_Ports). In the case of an Initiator or Target (Nx_Port), the port will grant some number of credits (typically between 8 and 64) to the switch. It does this by specifying the number of credits to be granted to the switch in the FLOGI frame. If the switch accepts the FLOGI request, the number of credits that the switch will be extending to the Nx_Port will be contained in the FLOGI Accept frame. An example of the login and credit initialization process is shown in Figure 99.

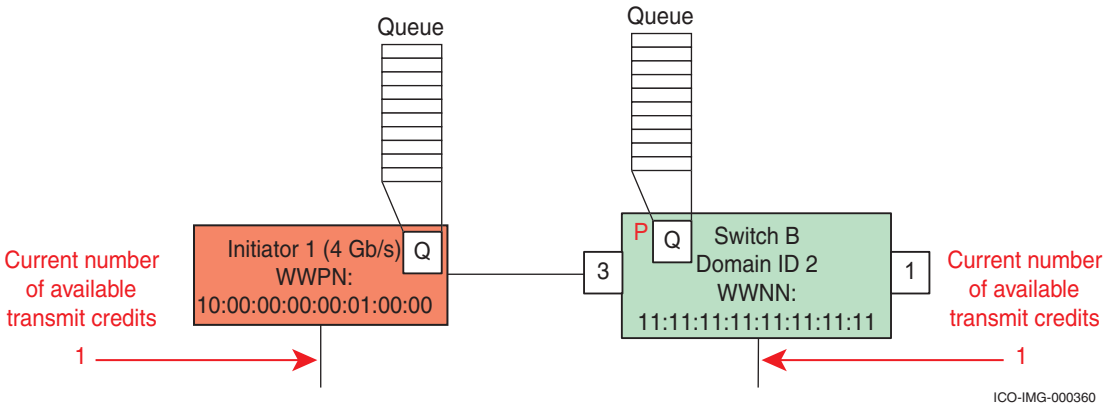


Figure 99 Login and credit initialization process example

Figure 99 shows an initiator and a Fibre Channel switch. Both the Initiator and the switch are shown as having a queue, each containing 10 buffers. Each buffer is capable of storing a single FC frame. All incoming frames will be stored in a buffer until they are ready to be processed by the switch or transmitted by the egress port. After the fiber cable is attached, the link will gain sync and eventually enter the ACTIVE state where the ports are transmitting IDLEs. After the ports are transmitting IDLEs, but before the FLOGI is transmitted, both ports have a single credit by default. This is so each port will be able to transmit a single frame, login, and negotiate credit.

"Receive" versus "Transmit" Buffer-to-Buffer Credits

Ideally, each port should keep track of two credit counts:

- The number of BB_Credits or buffers that it has remaining available to receive incoming frames (RX BB_Credit)
- The number of BB_Credits or buffers that are available on the port on the other side of the link (TX BB_Credit)

Each are further explained next.

- Receive Buffer-to-Buffer Credit (RX BB_Credit):

  The RX BB_Credit count is simply a count of how many buffers the local queue has remaining to accept incoming frames. When this number is zero, no additional Fibre Channel frames should be received. However, in practice this is not always the case. Sometimes due to a bug, the transmitter will send more Frames than for which it has been granted credit. This condition is called a *buffer overflow* and how it is handled is implementation dependent.

  Some ports, especially switch ports, will allow an overflow to occur and even route the FC frames received when RX BB_Credit is zero to their proper destination. They do this by keeping some number of spare buffers available for conditions like buffer overflow and storing these extra frames in these spare buffers.

  Other implementations will discard the frame and reset the link with LR (Link Reset Primitive) to re-initialize the BB_Credit counts to the login values.

  Both of these are acceptable, with the former being more forgiving and the latter being more rigid but closer to the spirit of the standard. In either case, as long as the transmitter is not constantly overflowing the buffer, both approaches work.

  If the transmitter is constantly overflowing the receive buffer, we enter a "works fine with vendor A (allows overflows) but not with vendor B (resets the link)" kind of interoperability problem.

  On the port on the other end of the link, the RX BB_Credit count is kept track of as the Transmit Buffer-to-Buffer Credit (TX BB_Credit) count.

- Transmit Buffer to Buffer Credit (TX BB_Credit):

  In addition to keeping track of how many receive buffers a port has left available, the number of receive buffers left on the port on the other end of the link needs to be kept track of as well. This is done through the use of the TX BB_Credit count. The count is

kept track of by the transmitter by decrementing the TX BB_Credit count by one for every frame it transmits and incrementing it by one for every Receiver Ready (R_RDY) primitive that it receives.

Receiver Ready (R_RDY)

The R_RDY is a 4-byte primitive sequence that indicates a buffer has been freed on the port that transmitted the R_RDY. It is important to note that a port that transmitted a frame and received the R_RDY is unable to determine which frame the R_RDY was sent for. Also, the R_RDY does not indicate that the frame was received in good condition or that it was forwarded successfully. It just indicates that a buffer on the receiver was freed and is ready to store another frame. The transmission of an R_RDY is also known as "releasing a credit."

BB_Credit loss

Occasionally, it is possible for the TX BB_Credit on a transmitting port to get out of sync with the RX BB_Credit on the receiving port. This typically happens for two reasons: Frame corruption or R_RDY corruption, each discussed next.

- **Frame corruption –** Due to a bit error on the link, a frame is corrupted in such a way so that it is not recognizable as a frame on the other end of the link. This is typically due to a corrupted **Start of Frame** delimiter. Since the receiving port does not know that a frame is being received, it would just detect non-frame data and depending on the implementation, handle it in a number of different ways. However, it would not release a credit since it did not get a frame.
- **R_RDY corruption –** Due to a bit error on the link, an R_RDY primitive is corrupted in such a way so that it is not recognized on the other end of the link. Since the port never receives an R_RDY, it never increments the TX BB_Credit count.

In either case, if the transmitter started with ten TX BB_Credits and then decremented this count by one after sending the frame, the counts would be out of sync since the transmitter would think it only had nine TX BB_Credits left while the receiver would think that ten were still available. If this were to happen often enough, a transmitter could lose most of the TX BB_Credit and poor performance, particularly over distance, would result. See the "Poor performance example," next, for an actual case study with a recommendation how to detect and resolve the problem.

Poor performance example

Ignoring the long roundtrip times in distance applications and the error recovery mechanisms that may be invoked by the initiator or

target, which are by no means trivial, a single bit error can have a surprisingly large impact on the throughput rate. For example, take a 4 GB/s link with a single bit error. Many host and storage ports will re-drive an entire exchange upon a single bit error, so for the sake of this example, assume that it is a 64k write (usually 32 2k frames). In this example, you would need to re-send these 32 frames, which means that during the simple second they are being transmitted, your maximum data transfer rate is 412.25 MB/s minus 0.064 MB/s = 412.17 MB/s.

This may sound insignificant, but when you consider that 1 bit represents 0.0000000235% of the total bits transmitted in a second, and it cuts the throughput by 412.17/412.25 = 0.24%, you are talking about a difference of seven orders of magnitude. If you have 10 bit errors per second, your effective throughput will be decreased by 2.4%. If there are 100, it will be decrease by 24%. Again, this does not consider the impact that invoking error recovery mechanisms will have on the environment.

There is another more insidious problem when bit errors are being logged: should one happen in an R_RDY, the link will lose a credit that will not be recovered until the next Link Reset. To discover the odds of having a bit error impact an R_RDY, read the next section, "Buffer-to-Buffer Credit issue."

Buffer-to-Buffer Credit issue

As previously mentioned, a Fibre Channel link consists of a transmitter and a receiver. Each receiver includes a set of frame buffers. Buffer-to-Buffer Credit is the flow control mechanism that ensures transmitted frames will be received.

Whenever a transmitter sends a frame it decrements its credit counter. The transmitter will continue to send frames and decrement the counter until the counter is zero. When a receiver succeeds in releasing one of its buffers by moving the frame to the next port in the route, the receiver sends an R_RDY to the transmitter. The transmitter increments its Buffer-to-Buffer Credit counter for each R_RDY it receives.

When the link is short and the route is uncongested, the counter is expected to stay near its maximum value. When a link is congested, this value may fluctuate heavily and spend time at zero credit. This is a normal and expected behavior of a Fibre Channel environment. As soon as the congestion is relieved, the frames are forwarded and

Credit is recovered. The source of this backpressure may be a few hops away, originating with an N_Port.

What can cause the Buffer-to-Buffer Credit issue to trigger?

Performance of an FC link depends on the BB_Credit counter being incremented by the receipt of an R_RDY for every frame it transmitted. If an R_RDY is not received, the counter does not increment. If this happens repeatedly, the credit will be decremented to zero and the link will stop functioning.

In a DWDM environment servicing a distance application, BB_Credit is necessary to keep the link fully utilized. The loss of credit can lead to ìdroopî, a condition where there is insufficient credit to support the bandwidth of the link. A 30 km 2 GB/s link will start to droop if available credit falls below 30. The link will continue to operate but at reduced performance. This credit leakage scenario can be caused by two events:

- R_RDYs are corrupted.

  One side thinks credit has been given back, but the other side never receives the credit. The credit counter will not be incremented.

- One side sends a frame and decrements its credit, but the other side never receives the frame.

  Under this scenario the receiver will not launch an R_RDY. The same effect occurs if the SOF (Start_of_Frame) field is corrupted.

The R_RDY and SOF field represent a small percentage of the consumed bandwidth. R_RDY and SOF are 4 bytes each. A typical data frame will consume 2104-bytes. Given the corruption is randomly distributed, for every corrupted R_RDY or SOF frame you would expect 263 data frames (2104/8) to have been corrupted. For every data frame corrupted, the FC-SCSI Exchange will have to go through an error recovery that can significantly impact response time.

In the case of SRDF, there is a three-second recovery timer. An/ IO that was supposed to be completed in 10 milliseconds or less could see a 300x slowdown for each corrupted exchange. In order for a 30 km link with 60 credits to lose 30 credits and cause the link to drop, it is likely 7,890 frames (263x30) will have been corrupted and each one will have impacted response time.

As a result, it is best to have a link quality monitoring strategy that monitors overall frame integrity. Monitoring or automatically recovering BB_Credit would mask the real problem, subjecting the environment to ongoing response time penalties. In addition it is very difficult for the switch to distinguish between lost credit and normal congestion events in a timely fashion.

How to resolve with minimal intrusion:

EMC recommends a two-prong monitoring strategy:

- Characterize the nature of the corruption events on both sides of the link by using a monitoring tool to monitor and record link quality metrics.
- Implement an alerting system on both sides of the link to trigger before the risk of a business impact is high.

The most sensitive means of tracking link quality is with the Invalid Transmission Word (ITW) counters in the switch.

EMC also recommends setting up two Counter Threshold Alerts, one for short bursts and one for multiple single occurrences distributed in time.

- The short burst alert would be 40 ITWs in 5 minutes.
- The single occurrence alert would be 100 ITWs in 24 hours.

These values are an initial recommendation and may need adjusting to avoid false alarms. These values are expected to trigger long before 30 credits are lost and performance is impacted.

The performance impact level calculated earlier in this section was 7,890 corruption events. The thresholds of 40 and 100 should provide adequate early warning. A third trigger could be created with a threshold of 4000 ITWs. When encountered, this would indicate to the user that they should manually execute a Port Reset to recover any lost credit.

The most critical statistics to monitor are:

- Invalid Transmission Words
- Class 3 Discards
- CRC Errors

Although setting thresholds and monitoring links for bit errors can help detect the loss of credit, it would be helpful if two devices were able to detect this condition themselves and recover from the loss of a credit. Recently, a few vendors have started to implement BB_Credit Recovery, which can help to do just that.

BB_Credit Recovery

BB_Credit Recovery is defined in FC-FS-2 and is only implemented on a few of the products that E-Lab currently qualifies. It is expected to be very useful in long distance applications. It works by sending special primitives at intervals that are agreed upon at login. The intervals are defined by the number of frames transmitted and the number of R_RDYs transmitted. If a mismatch is detected then an appropriate number of R_RDYs are transmitted in order to resolve the mismatch. A port specifies that it is capable of supporting BB_Credit Recovery at login time.

Login and credit initialization

Note: To reduce the number of diagrams that would otherwise be needed in this section, vertical bars will be drawn under each SAN component and horizontal (slightly diagonal) arrows will be inserted between the vertical bars to indicate the transmission of a frame between components. Individual frames, such as the FLOGI, will not be shown wherever possible.

As shown in Figure 100, once the initiator transmits the FLOGI, it will have zero transmit credit left and will be unable to send any other frames. An accept to the FLOGI is needed before the initiator can use the fabric.

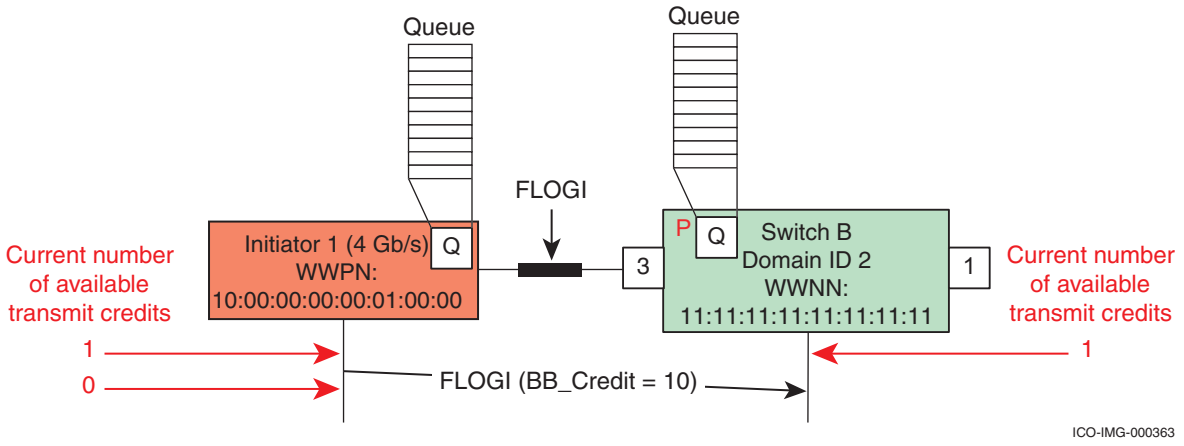


Figure 100 Login and credit initialization

As shown in Figure 101, once the switch receives the FLOGI, it will store it in a buffer until the switch is ready to process it. The processing begins when the FLOGI frame is forwarded to the Fabric Server logic within the switch.

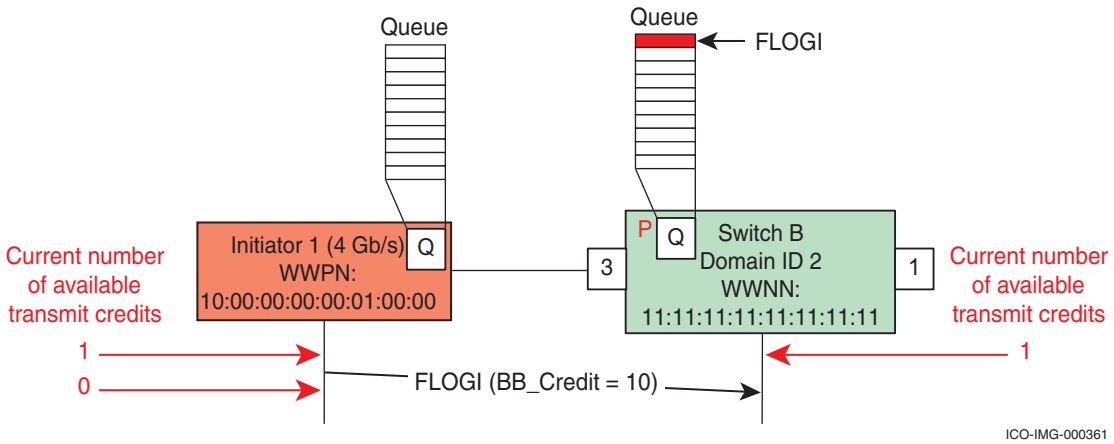


Figure 101 Switch receives the FLOGI

As shown in Figure 102 on page 229, once the switch processes the FLOGI, it will have 10 transmit credits to send frames back to the initiator with. Next, the FLOGI Accept frame is transmitted by the switch, which decrements its number of transmit credits by one. Once the FLOGI Accept frame is received by the initiator, it will update its transmit credits to the value in the FLOGI Accept and then it will release a credit by transmitting an R_RDY.

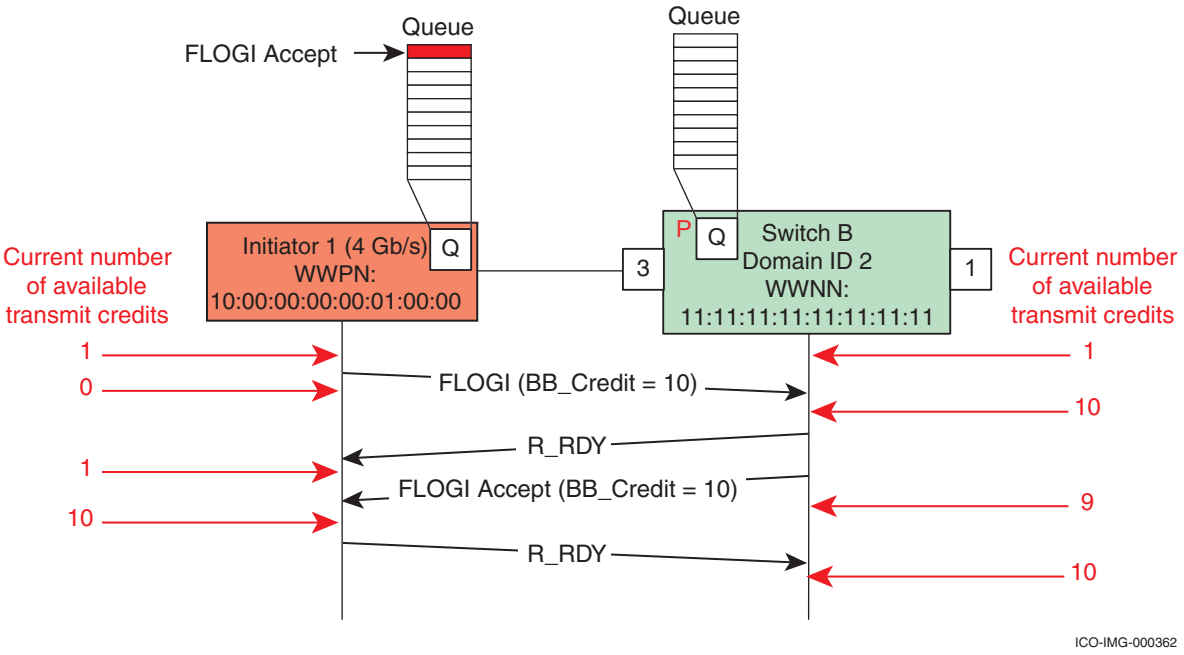


Figure 102 Switch processes the FLOGI

Queue types

In Figure 103, Initiator 1 is attached to Switch B. Above Switch B is a *queue*. The queue is where any incoming frames are stored until they are ready to be transmitted. Each queue has 10 buffers and each buffer is capable of storing a single FC frame.

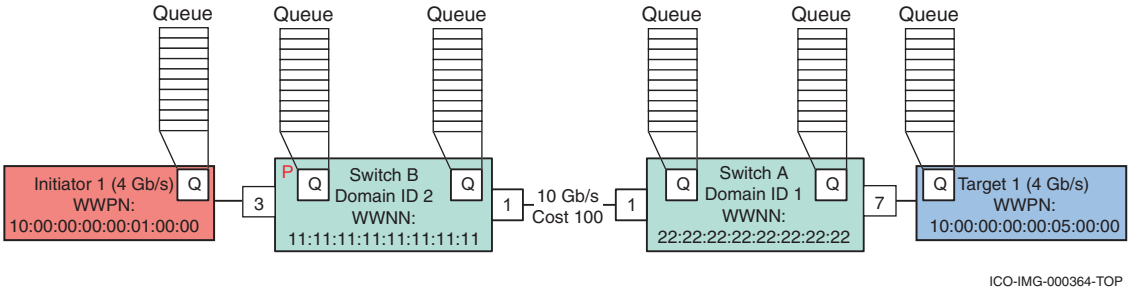


Figure 103 Queues

For the purposes of this discussion, we will consolidate the different queuing mechanisms in use into two groups:

- Shared memory
  Shared memory buffers lump all of the buffers used by a switch or set of ports into one common buffer area and then the transmitting ports read frame data from this buffer area as they are transmitting the frame.
- Virtual Output Queues

  Virtual Output Queues are typically created at a switch ingress port. They work by grouping frames for a given switch egress port (output port) on the switch into the same virtual queue and then these frames are transmitted in the same order they were received. This ensures that frames will not be delivered out of order while at the same time ensuring that there are no head of line blocking problems.

  Virtual Output Queues have finite resources so, although Virtual Output Queues to a given egress port can grow and shrink independently of one another, they are limited to the total number of buffers available in the Queue. In other words, it is possible for the Queue to be completely consumed by frames all destined to the same egress port. The other Virtual Output Queues will still be there but there will be no space left available in the queue.

The type of queue that is used by a switch can vary on a product-by-product basis. Each different type of queuing has it pros and cons but at some level they all do the same thing: *they store frames*. When a switch detects the Start Of Frame (SOF) delimiter, it stores the SOF, all of the data that follows (up to a certain amount, usually about 2 KB bytes), and the End Of Frame (EOF) delimiter into a queue (see Figure 104).

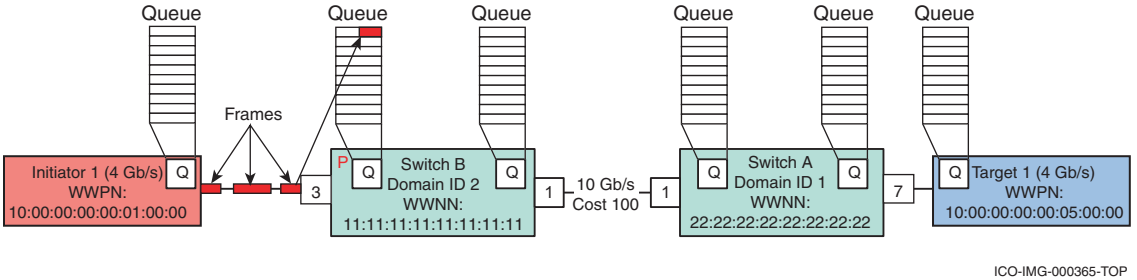


Figure 104 Queue example

What happens next depends on the type of queue being used. In any case, once a switch is ready to transmit a frame, the transmitting port will start pulling the frame out of the queue and begin transmitting it, as shown in Figure 105.

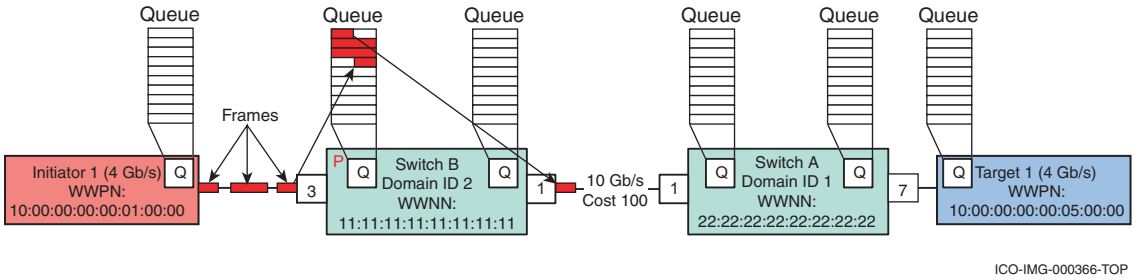


Figure 105 Transmit example

In Figure 105 there are parts of four frames in the queue. Depending on the switch architecture, when there is no contention for transmit resources (as implied above) it may be possible to immediately start transmitting the frame without needing to store the entire frame first. This is sometimes referred to as *cut through routing*.

Some switches that do not support true cut through routing do have low enough latency so that they are not considered *store and forward*. *Store and forward* architectures are those that may not start transmitting a frame until the entire frame has been received.

As shown in Figure 106, once Switch B has transmitted the frame, the buffer is freed and a credit is released via R_RDY.

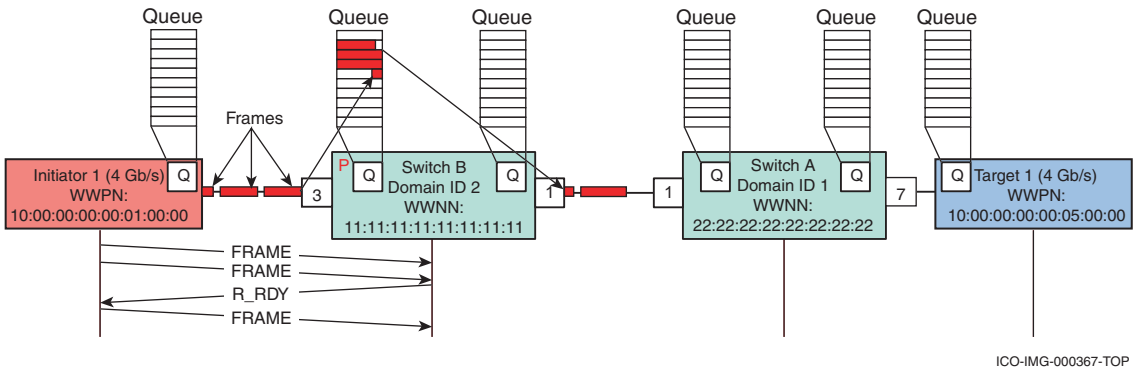


Figure 106 Released credit

Sources of backpressure and congestion

The example of a bus station was provided at the beginning of "Congestion and backpressure" on page 217, to illustrate the concepts of congestion and then backpressure. It was stated that as long as passengers left the terminal at the same rate as they entered, the transportation system would function properly. The same is true for a Fabric. As long as an average number of frames being transmitted from the switch is equal to the average number of frames being received by the switch, congestion and backpressure will be minimal. In some cases however, it is not always possible for frames to be transmitted as quickly as they are received and, as is the case with the transportation system, the Fabric will experience congestion. If the congestion persists long enough, backpressure will result. There are two primary reasons for congestion:

- A slow drain device
- Oversubscription

Both are further explained in this section.

Slow drain device

Occasionally, due to an architectural limitation, overall system load, poor volume layout, or a malfunctioning internal component, it is not possible for a destination Nx_Port to process frames at the same rate as they are being received from the fabric. An example would be a target port without enough cache.

As shown in Figure 107 on page 233, if the average number of frames being sent to a target port exceeds its ability to process those frames, Queues will start to fill up. Notice that the difference in transmission and reception rates does not have to be very large. A difference of 60 KB/s would be enough to cause the congestion scenario illustrated in Figure 107.

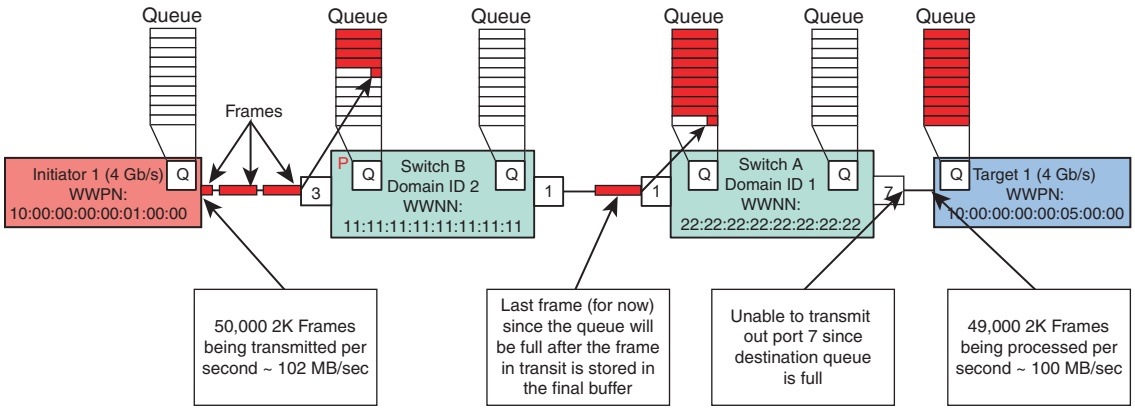


Figure 107 Slow drain example

In Figure 107, multiple points are experiencing congestion and backpressure. The congestion at target 1 is causing backpressure to be experienced by Switch A port 7 since it is unable to transmit. This in turn causes congestion at port 1 on Switch A, which results in backpressure at switch B port 1. Port 3 on switch B is not congested (yet) and as a result the initiator is not feeling any backpressure.

In this scenario, only the initiator and target being shown would be impacted by this mismatch but as will be shown shortly, this is not the case in a real-world environment containing multiple initiator and target pairs.

Two initiators and targets no congestion and backpressure

Figure 108 on page 234 shows an uncongested environment containing multiple initiators, each one transmitting to their own target.

Note: All Queues should be considered to have the same number of buffers even though they are not displayed that way in the illustration below. Although the Queue located near each port is intended to indicate a virtual output queue, the same type of issues can also be experienced in environments utilizing shared memory types of buffers.

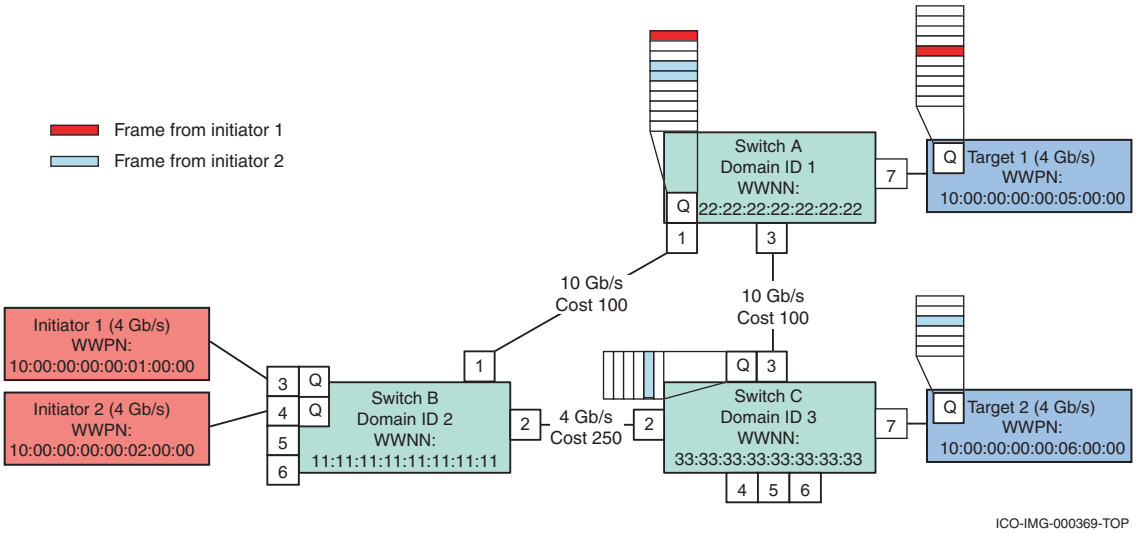


Figure 108 Uncongested environment

Two Initiators and one slow drain causes congestion or backpressure

Figure 109 on page 235 through Figure 111 on page 236 illustrate what can happen when a single slow drain device is present in a fabric. As shown in Figure 109, the queue on target 1 is full. This can happen for any number of reasons, as explained in "Slow drain device" on page 232.

Note: For the sake of this example, assume that both initiators are transmitting at the same rate but that Target 1 is handling frames at a rate that is less than they are being transmitted by the initiator.

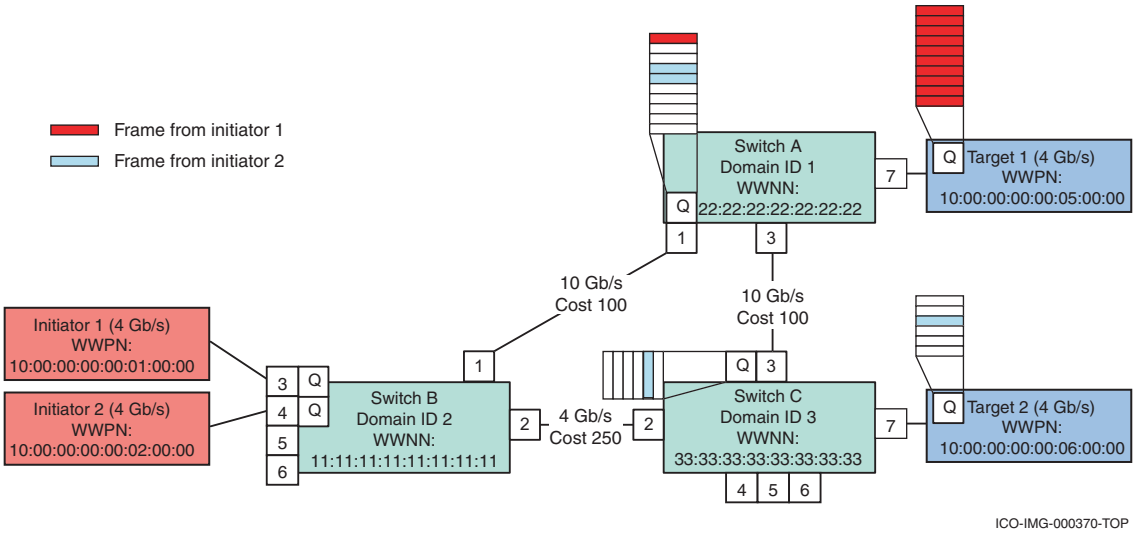


Figure 109 **Impact of a slow drain port**

The impact this slow drain port will have on the rest of the fabric will first be felt on Switch A, as shown in Figure 109. Since the Queue on Target 1 is full, port 7 will have no transmit credit and will be unable to transmit any of the frames in Switch A, Port 1's Virtual Output Queue for port 7. However, since the Queue on Target 2 is not full, little of the Virtual Output Queue for port 3 (within the Switch A Port 1's queue) will be consumed.

If this condition persists more and more of the total number of buffers on Switch A, port 1 will be consumed by the Virtual Output Queue for port 7 and fewer and fewer buffers will be available for port 3 (see Figure 110 on page 236).

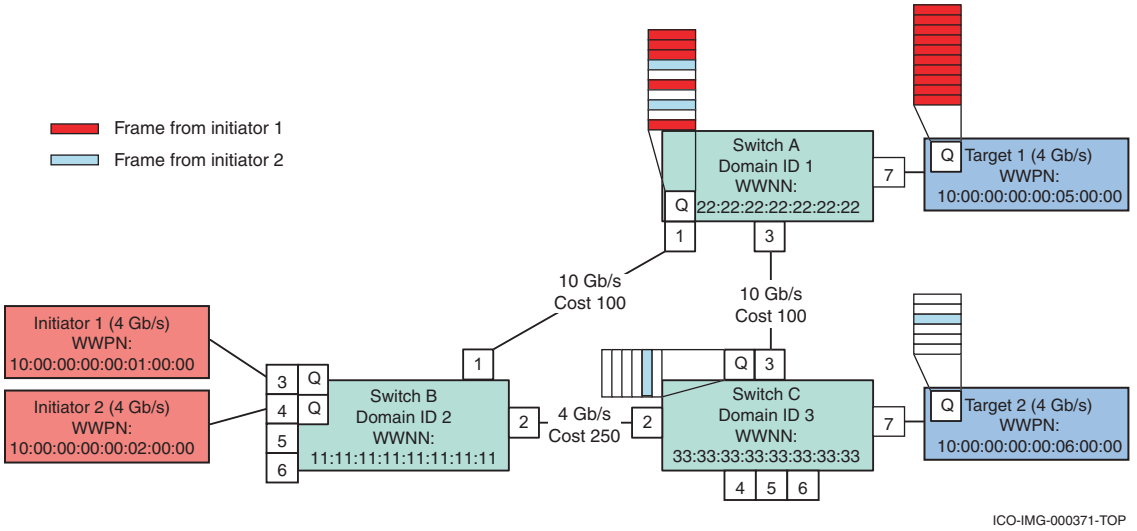


Figure 110 Virtual Output Queue for port 7 continues to grow

As shown in Figure 111, Switch A, port 1's Queue has been completely consumed by the frame destined for Target 1.

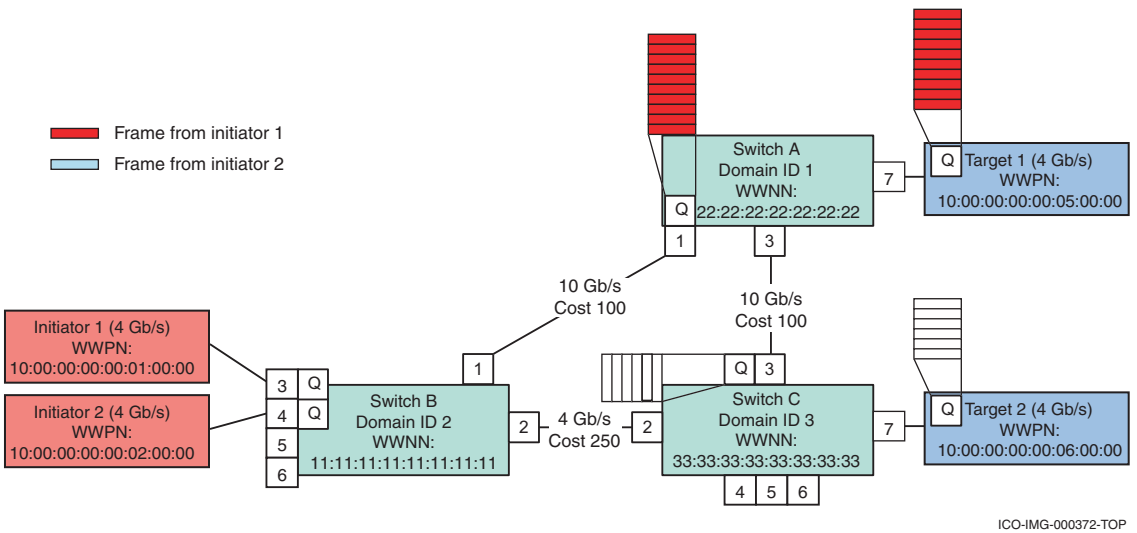


Figure 111 Virtual Output Queue for port 7 consumes entire Queue on Switch A, port 1

This condition cannot persist indefinitely for two reasons: the hold timer and the Link Reset Protocol, each discussed next.

- Hold timers

  The hold timer is the amount of time that a switch will allow a frame to sit in a queue without being transmitted. When the timer expires, the frame is discarded. Hold timers for Connectrix B Series, Connectrix M Series, Connectrix MDS, and QLogic are shown in Table 19.

Table 19 Hold timers

Vendor	Firmware	Hold Timer (milleseconds)
Connectrix B Series FOS	All	500
Connectrix M Series M-EOS	5.x and earlier	500
Connectrix M Series M-EOS	6.x	1000
Connectrix M Series M-EOS	7.x and later	1600
Connectrix MDS	All	500
QLogic	All	2000

This is not really a solution to our problem with the slow drain since average frame latency time through a switch is between 600 ns to 20 usec. The shortest hold timer above is 25,000 times greater than the longest average latency. The end result is that the hold timer has limited value for this particular scenario.

- Link Reset Protocol

Note: This is a worst case scenario and happens infrequently. Typically, a slow drain is only a slow drain periodically and this will result in occasional drops in total throughput. Finding a slow drain, especially without the use of a protocol analyzer, is one of the most challenging performance problems to troubleshoot.

If Target 1 does not release a credit within E_D_TOV (2 seconds), port 7 on switch A will transmit a LR (Link Reset) forcing the BB_Credit values back to the login values. Upon receiving the LR,

the receiver will discard all frames in the local queue. This will allow the queue on switch A, port 1 to be drained by Switch A, port 7 since it will now have transmit credit (at least temporarily). Since switch A, port 1 will now have buffer space, both Initiator 1 and 2 will be able to resume transmitting frames.

Oversubscription

There are a few different types of oversubscription that need to be considered.

- The first type tries to move more data across a link than it is capable of carrying. This is the classic case of having x capacity and needing 2x capacity.
- The second type is congestion due to a lack of BB_Credit. This is typical in a distance extension environment and can be architected around by proper planning. See the *Extended Distance Technologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab, for more information.

For the case where not enough capacity is available, it may be possible to architect around this issue through proper planning. However, when an ISL fails, as shown in Figure 112 on page 239, it is still possible to encounter congestion and backpressure.

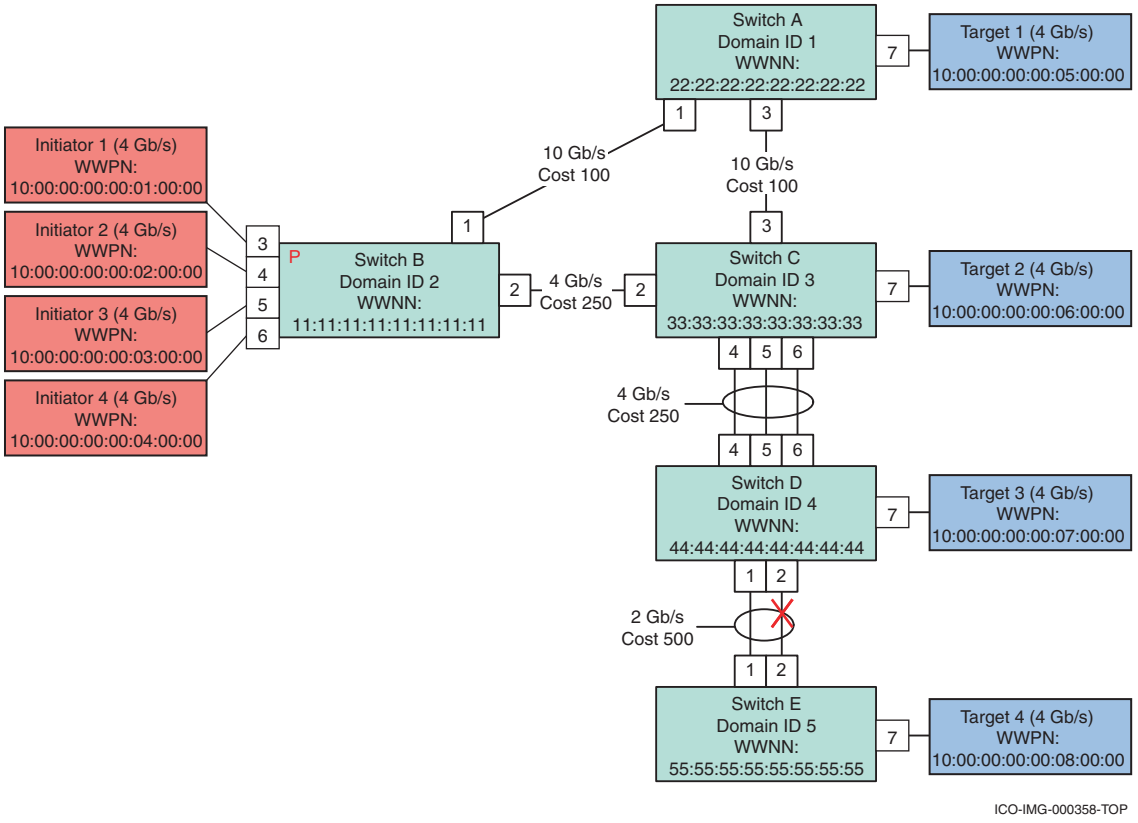


Figure 112 ISL failure

In Figure 112, an ISL has failed between Switch D and Switch E. If Initiator 4 was transmitting to Target 4 at something greater than 2 Gb/s, congestion would occur at Switch D, port 1 and backpressure would begin to spread back toward the initiator. As shown in the previous examples in this section, the impact this could have on the entire environment could be dramatic, so planning for this kind of failure is extremely important.

FLOGI

Once the link comes up to the active state, FLOGI (Fabric Login) is the first frame transmitted by a N_Port when it is attempting to log into a fabric. The N_Port uses the FLOGI frame to ask for permission to use the fabric and to request an FCID. The FLOGI request contains

the N_Ports Node WWN and Port WWN as well as the classes of service supported. It also contains the number of credits being extended by the N_Port to the switch.

The FLOGI ACC from the switch back to the N_Port contains the N_Ports FCID in the D_ID field, the switches WWPN, the WWNN, the supported classes of service, and the number of credits being extended by the switch to the N_Port.

There is no requirement that the BB_Credit values used by the N_Port and the switch be equal to each other. If an N_Port does not send the FLOGI within two seconds of coming up to the active state, the switch may attempt to initialize as an E_Port and transmit the ELP (Exchange Link Parameters) SW_ILS request. In this case, you could expect to see a segmentation reason for ELP failure or E_Port initialization failure.

Fore more information, refer to "Fabric Login (FLOGI)" on page 269.

Nodes

A Fibre Channel environment consists of two or more devices connected together by an interconnection topology. In Fibre Channel devices such as personal computers, workstations, disk array controllers, and disk and tape drives are all referred to as nodes. Each node is a source or destination of information for one or more nodes. Refer to Figure 113 on page 241 for an example. In EMC's case the node would be the Symmetrix system.

Each node requires one or more ports to provide a physical interface for communicating with other nodes through their ports. The port is a hardware attachment that allows the node to send or receive information using the physical interface. Some devices have these ports integrated and other have pluggable ports such as Host Bus Adapters for flexibility. In EMC's case, the port would be the port on the Symmetrix FA adapter.

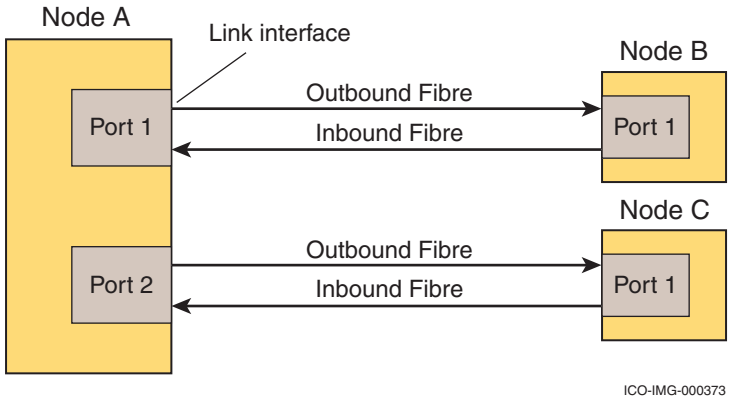


Figure 113 Nodes

Maximum hops

This section discusses how to count the number of hops in a Fibre Channel SAN, the importance of counting the hops, and highlights a number of important concepts, including the idea of logical versus physical hops.

Frequently asked question

What is the maximum supported distance between a host and its storage port?

Answer

In general, up to 10 km between a host and its storage port will be acceptable. The reason for the 10 km limit is because, as explained in "EMC support and recommendation" on page 242, the latency of the 10 km link happens to equal the latency of a typical switch that uses store and forward type of architecture. With this in mind, a 10 km link would just appear as another hop in the fabric.

Beyond 10 km, there is no quick and easy answer as to the maximum distance since it is completely dependent upon what the application can tolerate in terms of latency. Therefore, the best way to determine the maximum distance is to work with the Customer, determine the maximum latency acceptable to the application, and work out the maximum distance allowable based on the maximum latency tolerable.

In a Fibre Channel SAN, when a Fibre Channel frame crosses an ISL (Inter Switch Link), the frame is considered to have traveled across one *hop*, as shown in Figure 114.

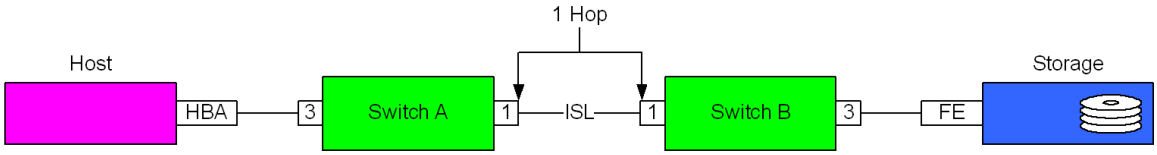


Figure 114 One hop example

As the number of switches in the SAN increases, it is frequently necessary to create a topology that will result in frames having to cross multiple hops to get from one side of the fabric to the other. The need for these extra hops is due mostly to the fact that topologies consisting of only one hop do not scale very well. Refer to "Common fabric topologies" on page 54 for more information on recommended topologies that minimize the number of hops in a given topology.

One feature that all of the recommended topologies have in common is that they do not exceed three hops.

EMC support and recommendation

At the time of publication, EMC supports up to five (5) physical hops (and the resulting logical hops) in the same fabric. The five hop limit is due to scalability and stability concerns.

Note: For additional information on logical versus physical topologies, refer to "Physical versus logical topologies" on page 28.

However, EMC recommends no more than three (3) physical hops in the same fabric.There are many reasons for the three hop recommendation, but the most important are discussed further in the following sections:

- "Performance" on page 243

  Performance is negatively impacted for each additional logical hop between a host and its storage port. Therefore, the number of logical hops should be minimized whenever possible.

- "Data integrity" on page 254

  Increasing the number of hops can increase the chances of data corruption occurring.

- "Fabric stability" on page 254

  Limiting hops in a fabric increases fabric stability.

EMC makes no restriction in regards to the number of logical hops between a host and storage port. This means that if both a virtualization appliance and a single FC Router are part of the topology, the number of logical hops could easily exceed 10, and be as high as 20.

Performance

One of the problems with crossing a large number of hops is that for each ISL crossed, the latency through the fabric increases. In addition, since I/O operations require at least a roundtrip through the fabric in order to complete, the roundtrip latency needs to be considered. As the roundtrip latency increases, the length of time each I/O takes to complete also increases.

SCSI READ

For SCSI **READ** commands, the length of the I/O increases by the length of time of the roundtrip latency. For example, a Read command requires that a command be sent and the data and status returned. When you include the latency each way, the Read data is delayed by at least the total latency time, as shown in Figure 115.

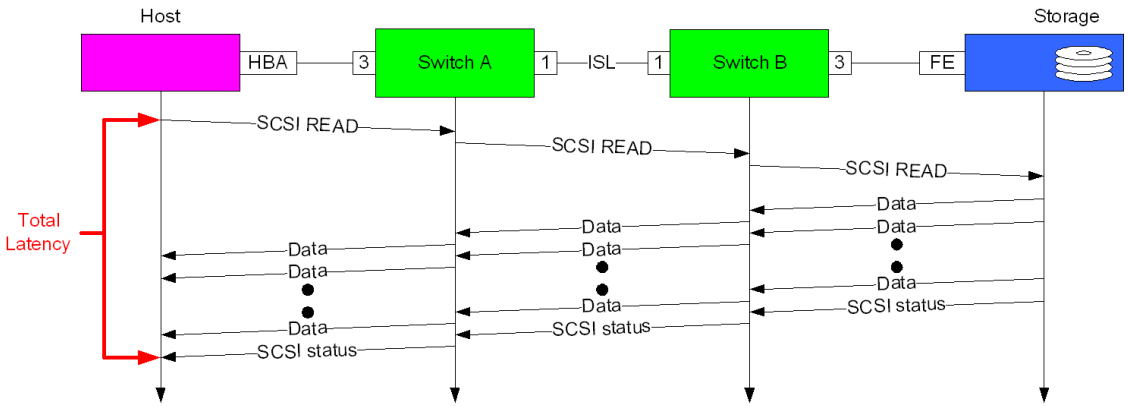


Figure 115 SCSI READ command example

SCSI WRITE

With **WRITE** commands, the **WRITE** command must be sent and the Storage port must respond with a transfer ready. The data will then be transmitted by the host and a status returned from the storage. As shown in Figure 116 on page 244, when you include the latency each

way with the WRITE command, the command completion time will be delayed by twice the roundtrip latency, since the command cannot be considered to be completed by the host until the status is received.

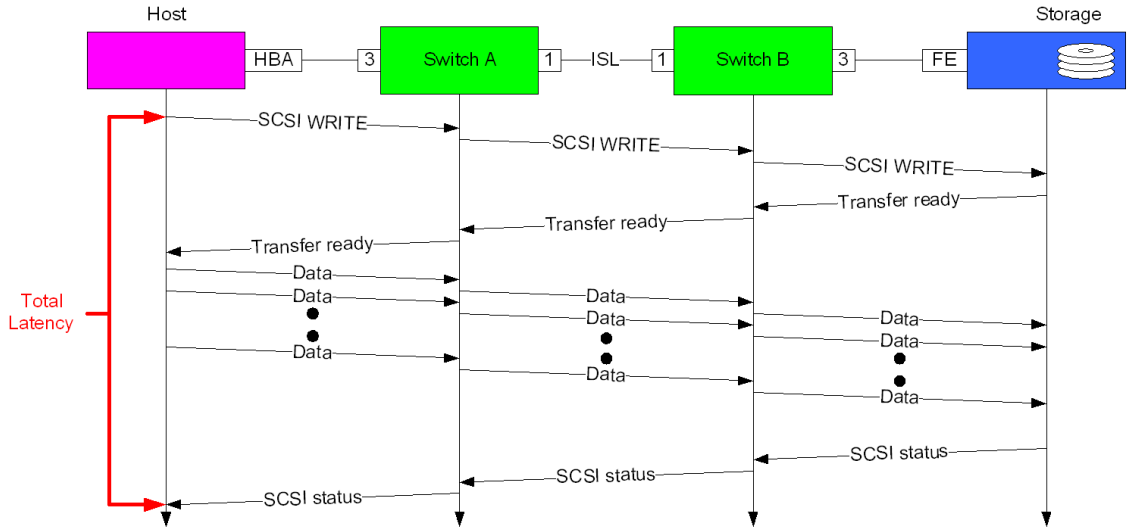


Figure 116 SCSI WRITE example

The total latency can be further broken down into *discrete* latencies such as Host, Link, Switch, and Storage latency. In Figure 117, a SCSI **READ** command is broken down into discrete latencies.

It is important to note that Figure 117 only shows a one-way latency since the Data and Status phases will also need to be performed.

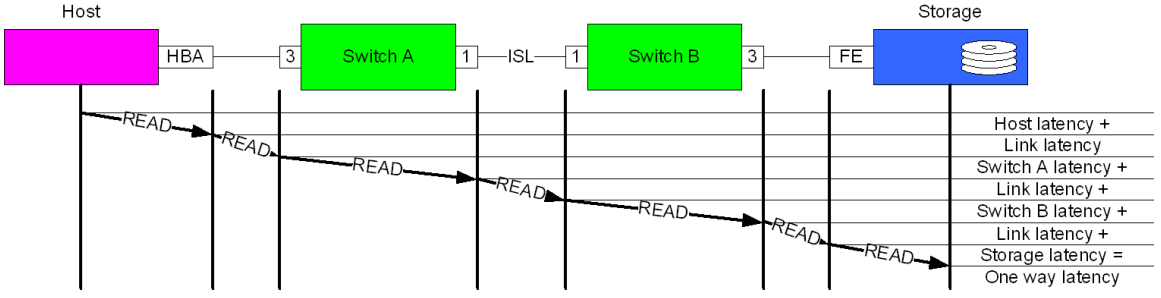


Figure 117 Discrete one-way latency example

The discrete latencies, Host, Link, Switch, and Storage, are further described as follows:

- Host latency — Host latency is the amount of time it takes for an I/O request from an application to make it from the application to the HBA transceiver. Typical host latency is less than a microsecond but factors, such as queue depth and number of transmit BB_Credit, can affect latency.
  - Queue depth is the number of outstanding I/Os a host can have with a particular logical unit (device). If the queue depth is exceeded, the storage device returns a queue full status in response to a command from the host, which impacts throughput. The exact effect of a queue full on a particular host is OS dependent.
  - Transmit BB_Credit is the number of frames the switch allows the host to transmit without receiving a response from the switch. Refer to "Buffer-to-buffer credit (BB_Credit)" on page 289 for more information on BB_Credit. When the host is unable to transmit due to a lack of BB_Credit, the fabric is congested and the host is experiencing backpressure. Refer to "Congestion and backpressure" on page 217 for additional information.
- Link latency — Link latency is the amount of time a bit spends on the fiber. In Figure 117, there are three separate Link latencies that will need to be accounted for. Fortunately, the calculation is fairly straightforward. However, before discussing the calculation, it is worth noting that there are numerous resources available on the internet that cover this topic in detail. It is also worth noting that the speed of light propagates down single mode fiber at a slightly different rate than through multimode fiber.

  The calculation divides the speed of light in a vacuum, "c," by the refractive index of the media that the light is traveling through. The speed of light is approximately 299,792,458 m/s and in the case of multimode fiber, the refractive index is approximately 1.538. This yields a propagation rate of:

  299,792,458 m/s / 1.538 = 194,923,574 m/s

  For our purposes, a more useful number would be to calculate how long it takes light to travel one meter or, put more simply, instead of m/s, we need to determine s/m. Using simple Algebra, we can conclude that 194,923,574 m/s can also be represented as:

  1s/194,923,574m = 5 ns per meter (approximate)

If you are familiar with the rule of thumb that 200 km = 1ms, then you know that this formula works out (at least approximately).

Now that you know how long it takes to travel one meter, you can figure out the link latencies for all of the links involved. Simply multiply the link length in meters by 5ns and you have the result. A simple table, next, shows some results.

Fiber length (meters)	Link latency (seconds)	Link latency (ns)
1	0.000000005	5
5	0.000000025	25
10	0.00000005	50
50	0.00000025	250
100	0.0000005	500
500	0.0000025	2500
1000	0.000005	5000
20000	0.0001	100000

As an example, refer to Figure 114 on page 242. If we assume that the HBA-to-switch and storage-to-switch connections are both 50m and the ISL is 20km, the total latency would be:

250ns + 100000ns + 250ns = 100.5 usec (one way)

- Switch latency — Fabric congestion and switch architecture both need to be considered when trying to calculate switch latency.
  - The first factor, fabric congestion, varies with how busy the fabric is. Refer to "Congestion and backpressure" on page 217 for additional information on how a fabric can become congested.
  - The second factor, switch architecture, will vary from one switch to another and varies from 600 nanoseconds to 20 microseconds under ideal conditions.

  Under fabric congestion situations, the maximum length of time a frame can be held by the switch is determined by the switches hold timer which varies from one switch to another and varies from 500ms to 2000ms. Refer to "Hold timers" on page 237 for more information.

As shown in Figure 118, in a 3-hop topology, since there are 4 queues that could potentially hold on to the frames for between 500 and 2000ms, the total roundtrip latency (taking into consideration only switch latency) could, in the worst case, be anywhere from 4 - 16 seconds.

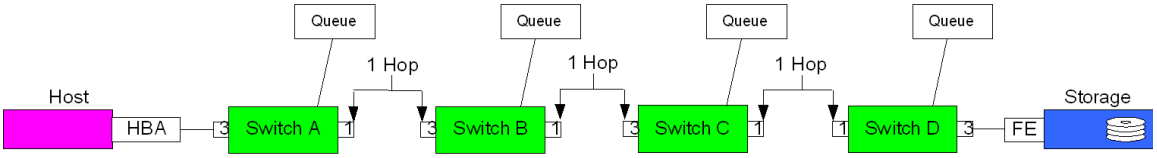


Figure 118 Switch latency example

It is important to note that if the switch latency were consistently anywhere near even a second, the applications running on the host would not be running for very long since applications and Operating Systems, in general, rely on sub-millisecond response time to operate properly. In fact, based on some testing performed in E-Lab, a simple two-switch fabric typically has latencies in the 30 - 40 microsecond range with latencies bursting as high as several hundred microseconds in heavily congested environments. Readers who are more knowledgeable with FC SANs will realize this is a gross generalization and is very dependent upon physical and logical topology. With this in mind, some real world data may be helpful, as shown in Figure 119.

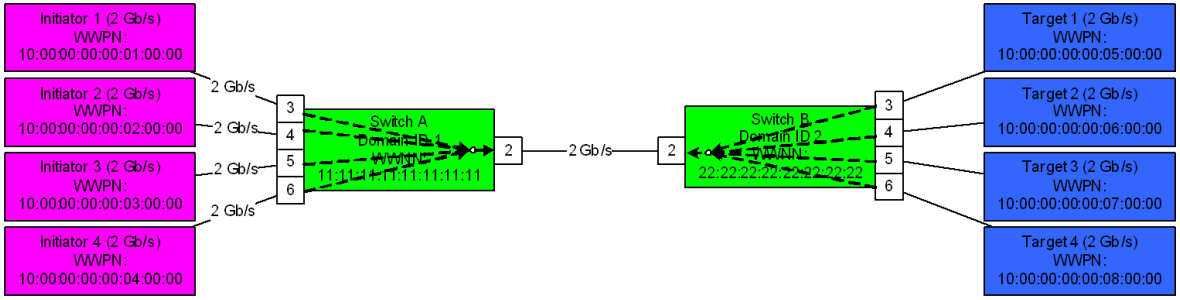


Figure 119 Simple two switch fabric latency example

In Figure 119, a simple two-switch SAN was created by connecting two switches with a single ISL fixed at 2 Gb/s. Four SANTester ports were connected to each switch. Each SAN Tester

port on Switch A attempted to drive I/O to another SAN Tester port on Switch B. This effectively forced all of the data across the ISL.

The average frame latency was then captured at several points, as shown in Figure 120.

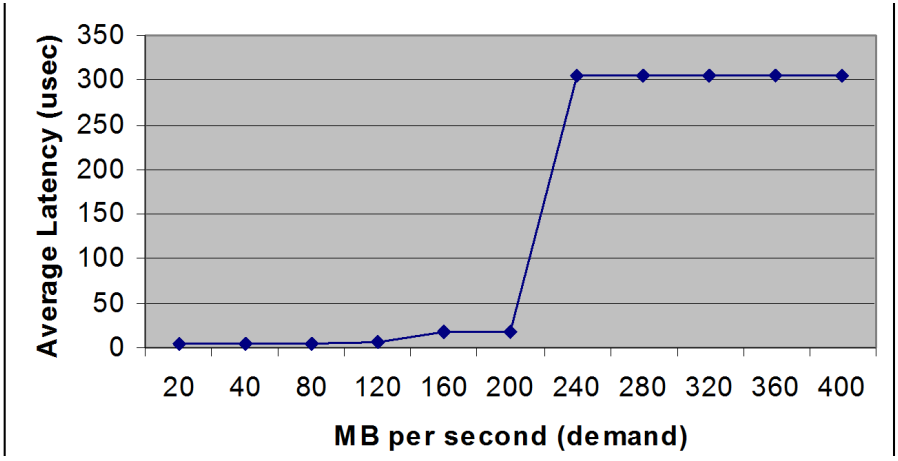


Figure 120 Throughput vs. latency

Notice that the average latency is much less than 30–40 usec up to the point when the ISL is saturated. (It is important to note that these numbers were observed to be about the same even when only one Initiator and Target pair were used.) Once the ISL was saturated, frame latency hit and then stayed at 300 usec while the effective throughput remained around 200 MB/s on the ISL.

- Storage latency — How long an I/O takes to be processed by a storage port depends on numerous factors. Generally speaking, an I/O will be handled by a storage port in much less than 10ms.

Impact on synchronous I/O operations

For the purposes of this section, a synchronous I/O will be defined as an I/O that must be completed before another I/O can begin. For those readers familiar with SRDF and Journal 0 (synchronous) mode, the idea of only having one outstanding I/O between a particular R1 and R2 device may be familiar. This ensures that every WRITE operation is replicated to a remote site before status is provided back to the host, and that a copy of the data exists at a remote location should, for any reason, the original become unavailable.

There is also the concept of a synchronous operation in which a particular operation, either within the OS or the application, needs to complete before other operations may proceed. Again, this is done to fulfill a particular need of the application.

Synchronous I/O or operations are particularly sensitive to latency. Many synchronous applications try to work around this problem by allowing multiple synchronous operations to happen in parallel, which works up to a point. Still, the fact remains, if the latency is too high synchronous I/O or operations will be the most severely impacted.

Assumptions

As an example, let us say we have a synchronous application that is only capable of performing one 64k write at a time, as shown in Figure 121.

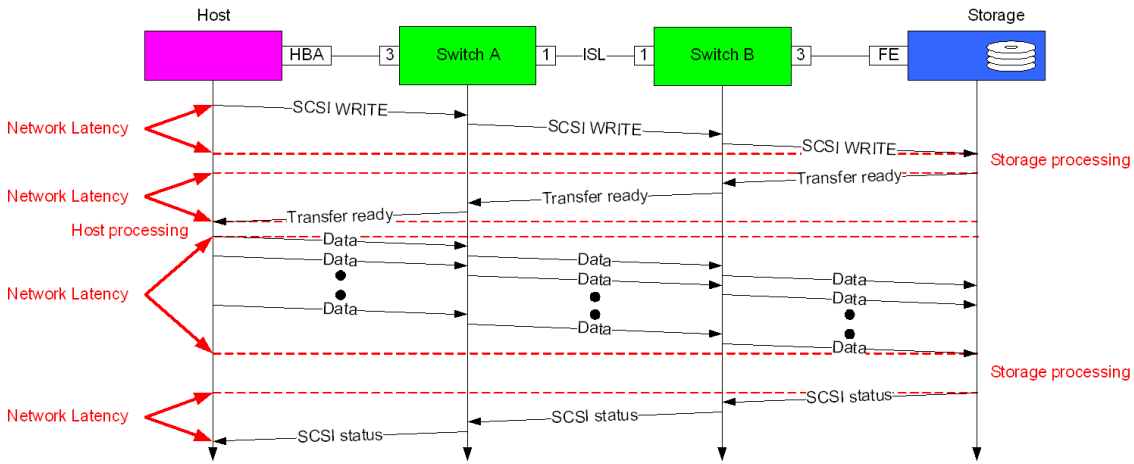


Figure 121 Synchronous I/O operations example

We will also assume the following:

- We have the proper number of credits to fully saturate the ISL between Switch A and Switch B.

Note: In this case since our application will only be sending 64k at a time, the maximum number of credits (2k buffers) that will be needed on the ISL is 32, regardless of the length.

- The fiber length from both the host-to-switch and storage-to-switch is 50 meters.
- The length of the ISL is 20 km
- There is no congestion in the fabric.
- There is a switch latency of 10 microseconds.
- There is a link speed of 4 Gb/s.

Data transfer time

The last assumption, link speed, does not actually impact when the first bit will arrive at the other end of the link, but it will impact when the last bit is received. This means that the slower the link, the fewer the number of bits that can be transmitted during the same amount of time as on a higher speed link. For the purpose of this section, this will be referred to as *data transfer time*.

Before we can determine data transfer time, the amount of overhead for the data to be transmitted needs to be taken into account. To do this for a SCSI **WRITE** command, we need to determine the size of the SCSI **WRITE** command, the transfer ready, the data frames, and the status frame. This is not time consuming since all four of these requests or responses are encapsulated in an FC frame.

As shown in Figure 122, each frame consists of an SOF (4-bytes), Frame Header (24-bytes), some amount of payload (x-bytes), CRC (4-bytes), EOF (4-bytes) and in the case of multiple frames being sent, the six primitive signals (24-bytes) that are required between each EOF and SOF.

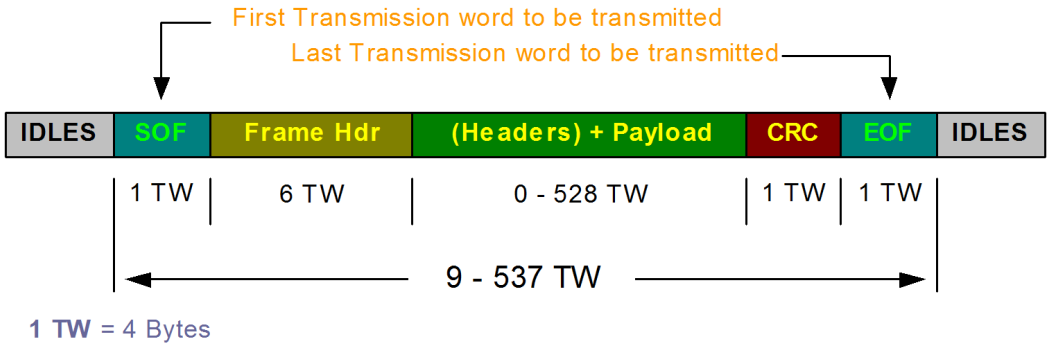


Figure 122 Frame format

For more information, refer to "Frame structure in Fibre Channel" on page 272.

In order to determine data transfer time, once the total amount of data to transfer, including overhead, is known, the number of *bytes* needs to be converted to the number of encoded *bits.* This is because of the 8b/10b encoding that is performed on all FC Frames transmitted. To convert bytes to bits, simply multiply by 8. To convert 8b data to 10b data, multiply by 1.25.

Once the number of bits to be transmitted is determined, multiply them by the bit time for the link speed you are dealing with, as shown in the following table.

Speed (Gb/s)	Bit time (ns)
1	1
2	0.5
4	0.25
8	0.125

In our example, since we are working with a link speed of 4 Gb/s, we multiply the number of encoded bits by 0.25ns.

WRITE(10) command

For a **WRITE** command, this works out to be 36 bytes of FC overhead and the SCSI FCP command payload, which is 32 bytes. A total of 68 bytes, or 680 encoded bits, will take 170ns (680*.25ns) to transmit.

Transfer ready

For a **WRITE** command, this works out to be 36 bytes of FC overhead and transfer ready of 12 bytes. A total of 48 bytes or 480 encoded bits will take 120ns to transmit.

Data frames

Each data frame contains 36 bytes of FC overhead, 2048 bytes of payload and, although not really part of the frame, we need to count 24 bytes of fill words if the data frame is not the last frame in the sequence. This is a total of 2084 bytes (20840 encoded) or 2108 (21080 encoded), counting fill words. Each frame takes 5.27usec to transmit (including fill words) except for the last one which takes 5.21usec to transmit, since the fill words do not need to be counted. Since 32

frames will be transmitted, the total time for the data to be transmitted will be 168.58 usec.

Status frame

The status frame contains 36 bytes of FC overhead and 12 bytes of payload. A total of 48 bytes, or 480 encoded bits, will take 120ns to transmit.

Total latency example

Taking everything into account, we can derive the total latency for the I/O and then determine the impact to a synchronous application.

On the first link, it will take 170ns for the WRITE(10) to be transmitted and then 250ns to travel the length of the fiber.

Add the switch latency we are assuming, which is 10usec.

Add 170ns for the WRITE(10) to be transmitted onto the ISL. Since the ISL is 20 km, the latency will be 100usec.

Add the 10usec for the second switch latency, 170ns for the WRITE(10) to be transmitted and then 250ns to travel the fiber to the Storage port.

This yields a total latency of:

170ns + 250ns + 10000ns + 170ns + 100000ns + 10000ns + 170ns + 250ns = 121.01usec

A similar process can be followed for the xfer ready, data and status frames. These calculations yield:

Transfer ready: 120.86usec

Data Frames: (total) 626.24 usec

Status: 120.86usec

If you add all of the latencies including the multiple data frames, the total latency due only to the fabric during the **WRITE** command is:

121.01 + 120.86 + 626.24 + 120.86 = 988.97usec

This means that during those 0.00098897 seconds, only 64k of data was transferred. If you look at this from a MB/s perspective, it works out to about 64.7 MB/s, which represents a significant impact to the application when you consider the link speed is capable of 400 MB/s.

Due to the 20 km ISL, the example might seem somewhat contrived to show the impact of latency on an environment. However, if you

refer to Table 20, derived using the same method as the example above, you will see that the number of hops has a significant impact on the total latency, and therefore on throughput.

Table 20 Total latency example (page 1 of 2)

Total number of hops	Average switch latency (ns)	Total link distance (m)	Link speed (Gb/s)	Total fabric latency (ns)	Host throughput (MB/s)
0	0	0	4	337980	189.36
1	600	150	4	514770	124.33
2	600	150	4	686160	93.27
3	600	150	4	857550	74.63
4	600	150	4	1028940	62.20
5	600	150	4	1200330	53.32
0	600	20100	4	742380	86.21
1	600	20100	4	913770	70.04
2	600	20100	4	1085160	58.98
3	600	20100	4	1256550	50.93
4	600	20100	4	1427940	44.82
5	600	20100	4	1599330	40.02
0	20000	150	4	420980	152.03
1	20000	150	4	669970	95.53
2	20000	150	4	918960	69.64
3	20000	150	4	1167950	54.80
4	20000	150	4	1416940	45.17
5	20000	150	4	1665930	38.42
0	20000	20100	4	819980	78.05
1	20000	20100	4	1068970	59.87
2	20000	20100	4	1317960	48.56

Table 20 Total latency example (page 2 of 2)

Total number of hops	Average switch latency (ns)	Total link distance (m)	Link speed (Gb/s)	Total fabric latency (ns)	Host throughput (MB/s)
3	20000	20100	4	1566950	40.84
4	20000	20100	4	1815940	35.24
5	20000	20100	4	2064930	30.99

Note: In the first entry with zero for hops, switch latency and link length is included to show that the time to transmit the command, transfer ready, data, and status have an impact on throughput and should therefore be considered the best possible throughput.

Data integrity

As discussed in "In order delivery" on page 262, frames arriving out of order can cause data corruption under special circumstances. If IOD is left at the default setting of *off*, increasing the number of hops can increase the likelihood of these special circumstances occurring in a fabric. It is important to realize that these corruption scenarios are extremely unlikely to occur and that the penalty, in terms of dropped frames and increased latency, may be excessive to the point of making it worth the risk.

Fabric stability

Another reason to limit the number of hops in a fabric is to increase fabric stability by decreasing the amount of time it takes a fabric to complete the fabric configuration process. The fabric configuration process is discussed in detail in "Build Fabric (Fabric Configuration) process" on page 154.

For another worst case scenario, refer to the fabric shown in Figure 123, which has been configured in a ring topology.

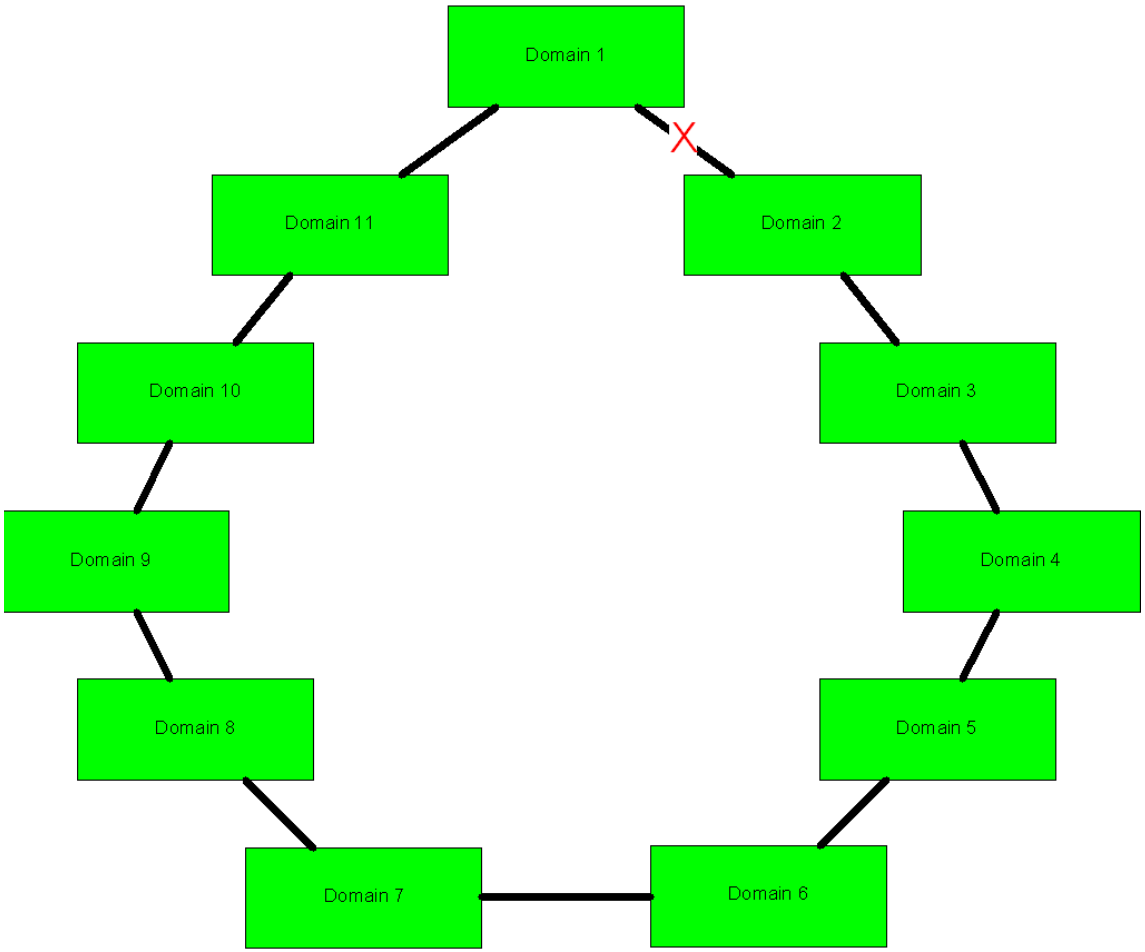


Figure 123 Ring topology example

In this topology, before the failure of the ISL between Domain 1 and 2, the fabric was in a topology supported by all currently available FC switch products because there were 5 hops or less between any two switches and less than 31 Domains.

After the ISL failed, there are 10 hops between Domains 1 and 2, which causes several problems. The first problem is that some switches will segment from the fabric when they detect a

configuration containing more than 7 hops. However, assuming that switches will not segment immediately, if the Build Fabric process were to begin for some reason (i.e., Domain 6 were to go offline and online), the process would take much longer than it would have had to if there were additional ISLs connecting each switch to more than just the switches it is adjacent to on the ring. In particular, refer to the "EFP – Exchange Fabric Parameters" on page 163, "Build Fabric (BF)" on page 164, and "Principal switch selection" on page 163.

Counting hops

Up to this point, we have assumed that the maximum number hops in the environment is the number of hops that all frames from the host will cross to get to the storage. This is not usually the case. In other words, we have assumed that the number of *physical* hops in the environment equals the number of *logical* hops between the host and its storage port. Refer to "Physical versus logical topologies" on page 28 for more information.

Depending on the environment, the number of logical hops may be less or greater than the number of physical hops. An example where logical hops are less than physical hops is when a host and storage are on adjacent switches and the switches are part of a fabric that has 3hops in it. In this case the number of logical hops would be 1.

An example where logical hops are greater than the number of physical hops could be when a virtualization appliance is used as in the example shown in Figure 124.

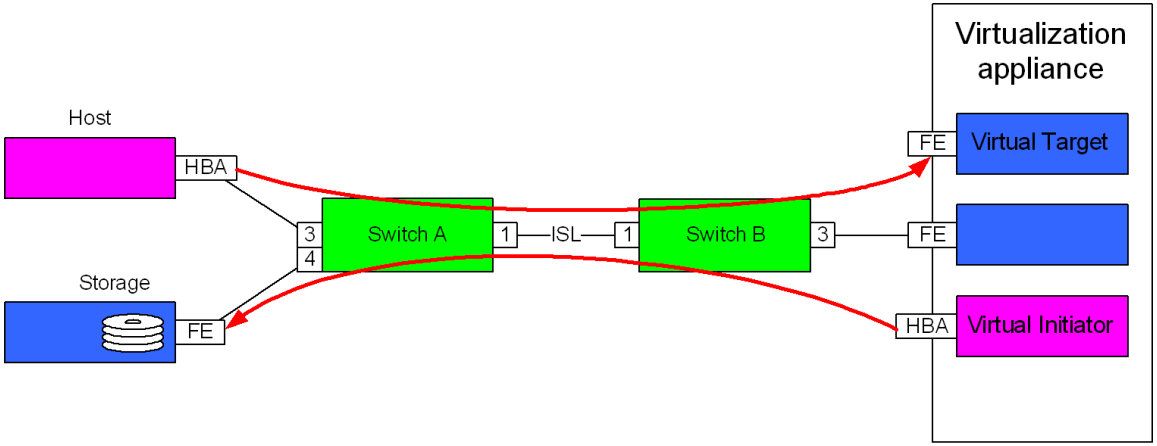


Figure 124 Logical hops are greater than physical hops example

The fabric shown in Figure 124 has one physical hop, but there are two logical hops between the host and storage. The first logical hop is from the host to the virtual target; the second logical hop is from the virtual initiator to the storage.

Note: When counting the number of hops in an environment, use the number of physical hops. However, always keep in mind the number of logical hops.

Another example where counting hops can be confusing is in an environment containing Fibre Channel Routers. Refer to the topology in Figure 125.

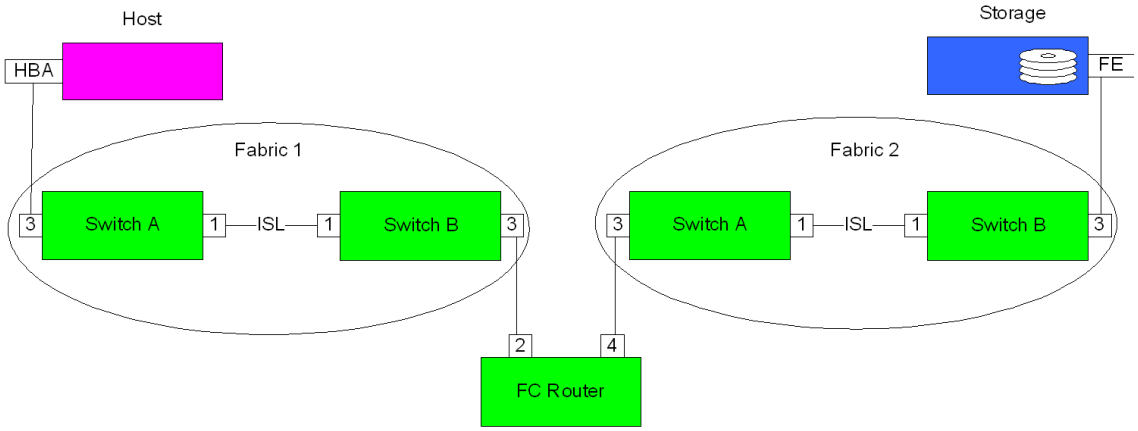


Figure 125 Fibre Channel Router environment

In Figure 125, each fabric contains two hops. In fabric 1, the hops are from Switch A to Switch B to the FC Router. In fabric 2, the hops are from the FC Router to Switch A to Switch B. The number of logical hops between the host and the storage is 4.

Flow control

When a port wants to send frames to another port the frames are sent from a buffer at the sending port and received into a buffer at the receiving end. However while a frame is being received at the receiver end if another is transmitted at the sender side then more than one buffer is needed to handle this at the receiver side. If there is no receive buffer space then that port cannot accept any more frames and this leads to busy responses and the possibility of frame loss.

To prevent this type of behavior and regulate the availability of receive buffers Fibre Channel uses two different flow control mechanisms to pace the rate at which the sender is allowed to transmit frames. The first is a link level flow control and is called Buffer-to-Buffer flow control. The second is a source and destination based flow control named End-to-End flow control (refer to "End-to-End Credit" on page 258). Both of these methods of flow control use a credit-based agreement between nodes to regulate the flow of frames between them. The credit value is the number of frames a receiving port has allowed a sending port to transmit to it.

For more information, refer to "Buffer-to-buffer flow control" on page 221.

End-to-End Credit

End-to-End credit (EE_Credit) is the maximum number of data frames a source port can send to a destination port without receiving an acknowledgement frame (ACK). This credit is granted during

N_Port login and is replenished with the return of an ACK response frame, as shown in Figure 126.

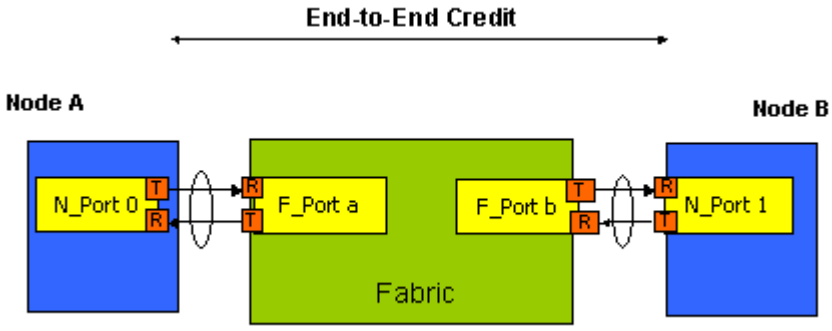


Figure 126 End-to-End Credit

EE_Credit is between the source and destination port and is only used in Class 1 and Class 2 circuits that are further described in"Class of Service (C.O.S.)" on page 284.

Figure 127 shows a summary of the mechanics of flow control and the main points to be considered for both End-to-End and Buffer-to-Buffer. It should be noted that Symmetrix operates in a class 3 circuit and thus uses Buffer-to-Buffer flow control.

	End to End flow control	Buffer to Buffer flow control
Regulate frame flow from	Source to destination N-Ports.	Transmitting N/F port to receiving N/F port over a link.
Class of service	1 , 2	2 , 3
Credit Counter	EE_Credit	BB_Credit
Signal	ACK	R_RDY
Frame Type	Data Frame only	All frames

Figure 127 Summary of flow control

Error detection and recovery

In Fibre Channel, error detection occurs at a number of different levels, further discussed next:

- At the FC-0 level errors associated with signal quality where primitive signals are used to reestablish link integrity.

- At the FC-1 level where it checks for invalid transmission characters or disparity issues and decides whether the frame is valid or invalid.
- At the FC-2 level frames are checked for various conditions such as CRC error detected, invalid transmission word detected, or the case where no receive buffer is available for the frame.
- At the FC-2 level when all active sequences have had a sequence time-out, a link time-out is detected, and a Link Reset is initiated.
- At FC-2 where during transmit or receive of a sequence the sequence initiator or recipient has detected an error and an action needs to be taken. For example detecting missing frames in a sequence or corrupted frames. Corrupted frames are discarded and the resulting error is detected and recovered at the sequence level. A missing frame is detected at the receiver by missing SEQ_CNT values and at the initiator by a time-out value being exceeded (E_D_TOV).

Time-outs provide a mechanism to detect where a certain operation did not occur within a certain time or did not occur at all and prevent ports hanging indefinitely. For example, if an HBA sends a **WRITE** command to the Symmetrix FA and it does not send a transfer ready back then after a certain time-out period an error will be detected and the Upper Layer Protocol will initiate recovery actions. Two of the most common time-out values are Error_Detect time-out value (E_D_TOV) and Resource Allocation time-out value (R_A_TOV).

E_D_TOV

The E_D_TOV time-out value is the timer for transmission of consecutive data frames and responses at the sequence level. Basically this is a short value and indicates how long a sequence can take to complete. It is a time-out value for communicating between two N_Ports that is negotiated at login time. Typical value for this is 2 seconds.

As can be seen in Figure 128 on page 261, if, for example, sequence 2 was a Class 3 sequence and the correct number of frames were not received at node 1 (frame 2 missing) a time-out period of ED_TOV will be allowed before a missing frame error is flagged. As this is Class 3 which has no Acknowledgements or Rejects it is up to the sequence recipient (node 1 in this case) to flag the error and it can then discard all frames in that sequence. Once all the frames in the sequence have been discarded it can then update the Upper Layer Protocol and this will notify the sequence initiator who will initiate

the recovery actions which can be to either to re-drive the entire sequence or exchange.

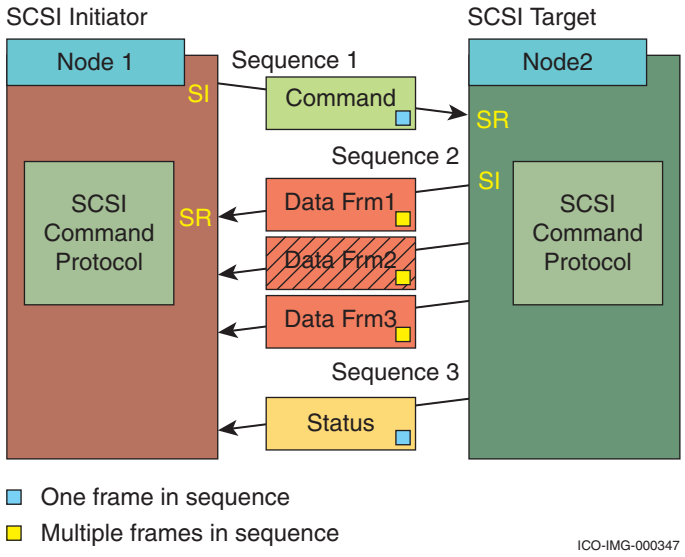


Figure 128 ED_TOV

RA_TOV

The R_A_TOV time-out value is a longer time-out value and defines how long before resources associated with an operation that has failed can be used again, for example, the time before a sequence initiator shall wait before it can reuse sequence qualifiers such as Sequence ID and sequence count. This value is basically how long an exchange can stay active until the ULP decides to retry the operation. A typical value for this would be 10 seconds.

In order delivery

This section discusses the following:

Environmental overview

Figure 129 shows a three-switch fabric. The host is in the middle of sending data to the storage, as would happen during a WRITE command. The blue boxes represent frames that are all part of the same exchange. The number inside each frame is the sequence count in hex. The majority of the frames are queued while waiting to be transmitted, as would be the case when the fabric is experiencing some form of congestion.

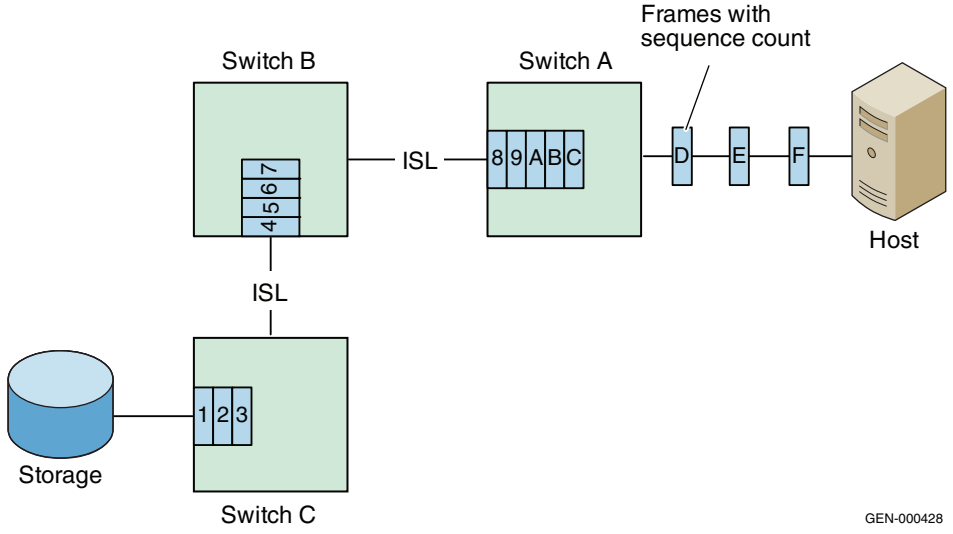


Figure 129 Host and storage separated by two hops

Addition of an ISL impact to FSPF cost

As shown in Figure 129, frames have to cross two hops to get from the fabric ingress port on Switch A to the fabric egress port on Switch C. Each hop has a cost associated with it. This cost is used by the

switches to calculate the shortest route from one point in a fabric to another. Determining the shortest path is the responsibility of the FSPF (Fibre-shortest-path-first) protocol.

For the sake of this example, assume that each of these ISLs is running at 4 Gb/s and has an FSPF cost of 250. This means that the total cost for the route from switch A to switch C is 500. Since there is only one route from Switch A to Switch C, it is also the shortest path. If another 4 Gb/s ISL were to be added between switch A and C and it had an FSPF cost of 250 as shown in Figure 130, a shorter path to switch C would then exist.

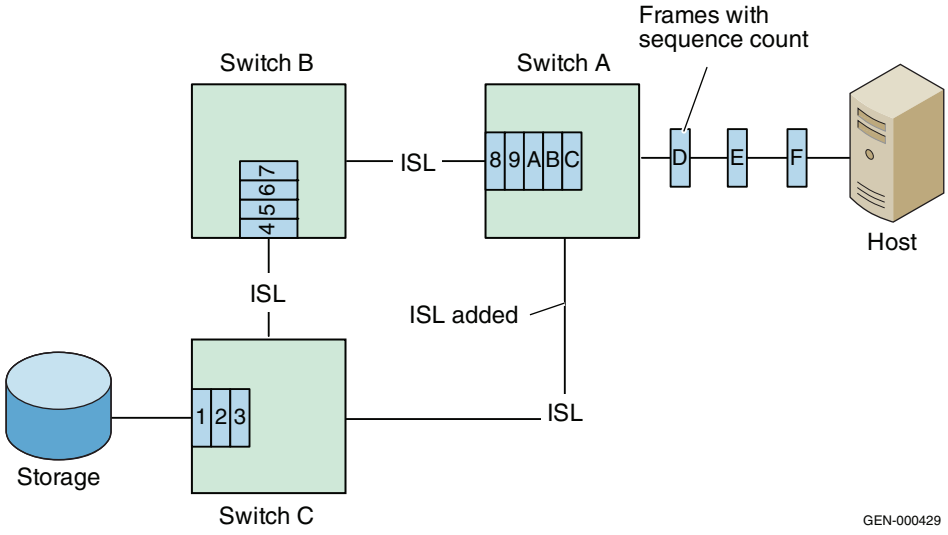


Figure 130 New ISL added. Shorter path introduced

The problem that the new shorter path introduces is that all frames received by the switch after the new ISL is established will take the new shorter path. If in-order delivery is not set on switch A, these new frames will be immediately sent down the shorter path as shown in Figure 131 and Figure 132 on page 264.

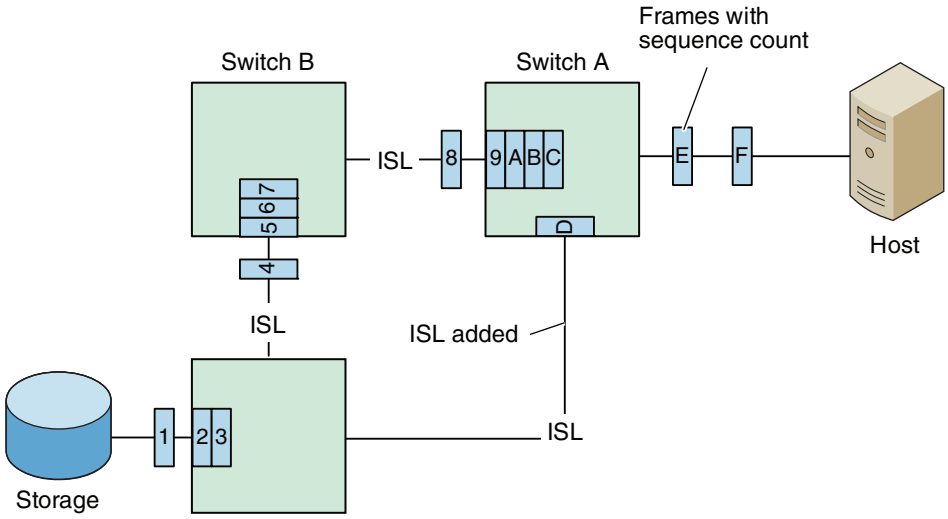


Figure 131 **New frames queued for transmit down shorter path**

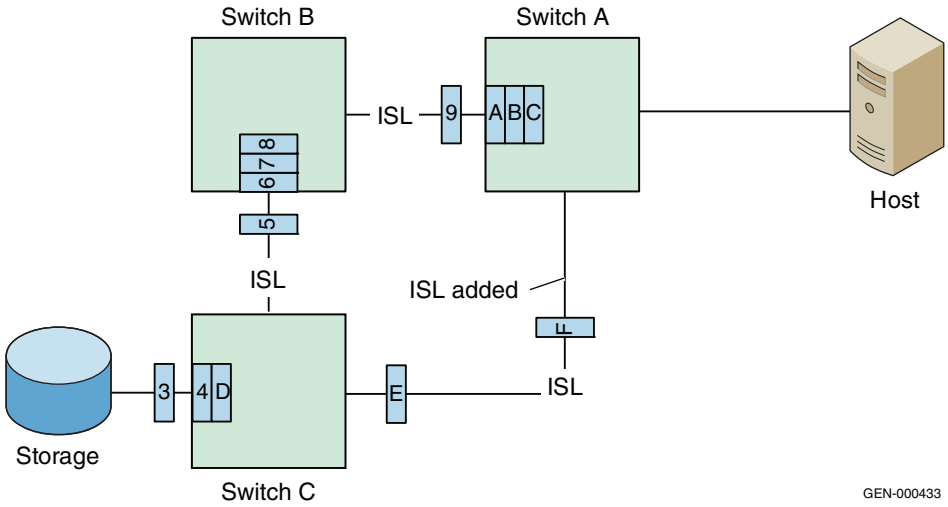


Figure 132 **Sequence ID "D" received before sequence ID "5"**

The problem with the frames being received out of order by the storage is that many hosts, and especially storage ports, do not support the re-ordering of frames. Instead, they will typically abort the exchange and force a retry.

If IOD is set, as shown in Figure 133, then new frames received will be held at the new ISL for some period (typically 2 seconds) while the rest of the frames on the older and longer path are allowed to drain.

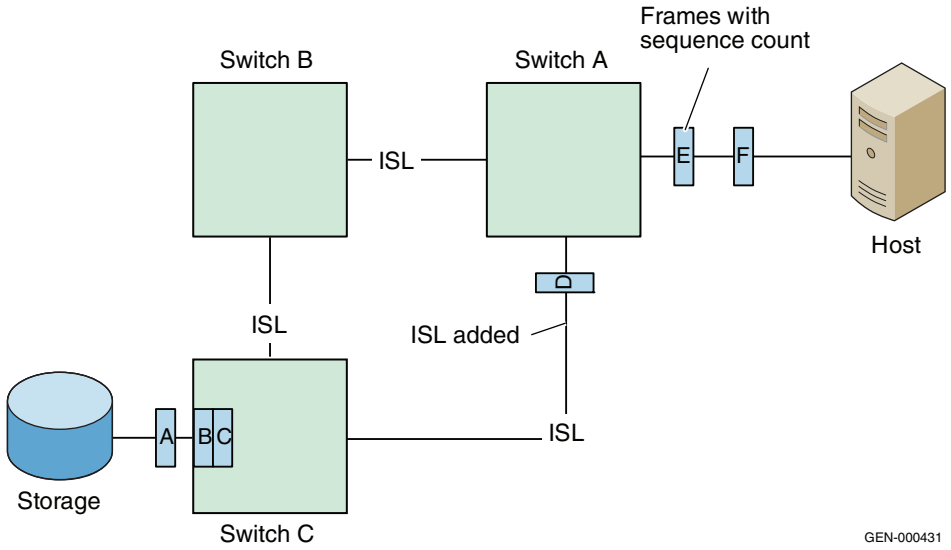


Figure 133 IOD is set. Sequence ID "D" is held and will be received in proper order

The advantage to setting IOD is that you are "guaranteed" that all frames will be delivered in order. The disadvantage is that you are guaranteed that all ports in the fabric that will use the new ISL will be paused for the hold time (typically 2 seconds). However, in most cases like the one described above, only a few exchanges will actually have frames arrive out of order and those exchanges are retried and completed well within the time it would have taken to guarantee that all of the frames arrived in order by setting IOD.

Inter-Switch Link (ISL)

An ISL (Inter-Switch Link) allows for two or more Fibre Channel switches to be connected together to form a single, but larger, fabric. For more information on how ISLs form, refer to "Build Fabric (BF)" on page 164.

Frame services in Fibre Channel

This section provides information on frame services in Fibre Channel:

Basic link service

Basic link service frames are link data frames that provide basic control functions within an exchange. These include:

- **No Operation (NOP)**. This basic link service has no data field and no meaning itself but is used to carry control bits between a sequence initiator and recipient.
- **Abort Sequence (ABTS)**. This is used to abort a sequence or an entire exchange.
- **Remove Connection (RMC)**. This is used to request immediate removal of a class 1 connection.
- **Basic Accept (BA_ACC)**. This is a response frame to a basic link service command indicating the command has been completed and the applicable information is being returned.
- **Basic Reject (BA_RJT)**. This is a response frame to notify the sender of a basic link service command that the command has been rejected for a certain reason. The reason code is supplied in the first four bytes of the payload.
- **Preempted (PRMT)**. This is used in Class 1 to notify an N_Port that the connection it is using has been removed.

Extended link service

Extended link services are used by an N_Port to provide a specific function at another port. They normally consist of a command (or request) sequence followed by a reply which ends the exchange. Examples of extended link services are Fabric Login (FLOGI), N_Port Login (PLOGI), and Process Login, each discussed in this section. For

a full list of the extended link services, see the table in Figure 134 on page 268.

R_CTL Type = 01	Payload Word 0 Byte 0	Name of Extended Link service	Abbr
22 Request	03	N_Port Login	PLOGI
	04	F_Port Login	FLOGI
	05	Logout	LOGO
	06	Abort Exchange	ABTX
	07	Read Connection Status	RSC
	08	Read Exhange Status Block	RES
	09	Read Sequence status Block	RSS
	0A	Requext sequence Initiative	RSI
	0B	Establish Streaming	ESTS
	0C	Estimate Credit	ESTC
	0D	Advise Credit	ADVC
	0E	Read Time-Out Value	RTV
	0F	Read Link Status	RLS
	10	Echo	ECHO
	11	Test (FCAL initialise commands)	Test
	12	Reinstate Recovery Qualifier	RRQ
	20	Process Login	PRLI
	21	Process Logout	PRLO
	22	State Change Notification	SCN
	23	Test Process Login State	TPLS
	30	Get Alias ID	GAID
	31	Fabric Activated Alias ID	FACT
	32	Fabric Deactivated Alias ID	FDACT
	33	N_Port Activated Alias ID	NACT
	34	N_Port Deactivated Alias ID	NDACT
	40	Quality od Service Request	QoSR
	41	Read Virtual Circuit Status	RVCS
	50	Discover N_Port Service Parm	PDISC
23 Response	01	Link Service Reject	LS_RJT
	02	Accept	ACC

Figure 134 Extended link services

As can be seen in Figure 134, there are quite a number of extended link services provided by Fibre Channel.

This section will expand on the following three services, as these are fundamental in understanding how the login process works:

- Fabric Login (FLOGI)
- N_Port Login (PLOGI)
- Process Login

Fabric Login (FLOGI)

Fabric Login (FLOGI) is used in Fibre Channel by an N_Port to detect the presence of a fabric and, if there is a fabric present, to establish a connection by exchanging certain parameters. The Fabric Login process also ensures the attached N_Port is assigned an address by the fabric. Fabric Login takes place following link initialization and is initiated by the N_Port sending a FLOGI extended link service to the switch well-known address 0xFFFFFE. At this time the N_Port does not have a valid fabric address and uses 0x000000 as the source identifier on the FLOGI frame. The N_Port sends its service parameters (operating parameters it supports) in the FLOGI frame and the *Accept* that is sent back by the fabric contains its service parameters and also a fabric address that will be used by that port as its source identifier.

Payload Word	
0	Command Code 04000000 FLOGI / 02000000 ACC
1 to 4	Common Service Parameters
5 to 6	N_Port Port Name
7 to 8	Node Name
9 to 12	Class 1 Service parameters
13 to 16	Class 2 Service parameters
17 to 20	Class 3 Service parameters
21 to 24	Reserved in FC PH Class 4 Service parameters in FC PH2
25 to 28	Vendor version level

Figure 135 FLOGI and Accept frame payload

Figure 135 shows the FLOGI and Accept frame payloads that will be seen during fabric login. The common service parameters and class specific service parameters are basically the FC-2 capabilities of the port.

A number of responses to the initial FLOGI frame other than Accept are possible. If the port receives F_BSY (fabric is busy, try again later) or F_RJT (the fabric did not like the FLOGI frame, class of service not supported or invalid S_ID) it may need to try attempt again to send the FLOGI frame.

N_Port Login (PLOGI)

N_Port Login (PLOGI) is used by an N_Port to send its service parameters to another port and to request the service parameters of the receiving port back in the Accept. The payload of the PLOGI request and the response (Accept) are identical except for the command code and is performed after fabric login and prior to any upper layer FC-4 type parameters are exchanged.

Payload Word	
0	Command Code 03000000 PLOGI / 02000000 ACC
1 to 4	Common Service Parameters
5 to 6	N_Port Port Name
7 to 8	Node Name
9 to 12	Class 1 Service parameters
13 to 16	Class 2 Service parameters
17 to 20	Class 3 Service parameters
21 to 24	Reserved in FC PH Class 4 Service parameters in FC PH2
25 to 28	Class 4 Service parameters
26 to 29	Vendor version level

Figure 136 PLOGI and accept payload

Figure 136 shows the structure of the payload of a PLOGI frame and the Accept. As mentioned in "Fabric Login (FLOGI)" on page 269, it is possible to have responses other than the Accept. The destination port could be busy in a fabric in which case a P_BSY response would be sent back from the N_Port or also the PLOGI could be rejected with a P_RJT if the class of service being requested was not supported by the receiving port.

Process Login

The Process Login (PRLI) is required to establish communication between two FC-4 layer processes (in our case SCSI) existing at two different N_Ports. The Process Login provides a way of allowing these two ports exchange SCSI information known as *service page* information. The Process Login is normally sent from initiator to target to establish SCSI FCP operating features between both. Again, the mechanism used is for the initiator to send a PRLI request frame and the corresponding Accept contains the service page information on the target.

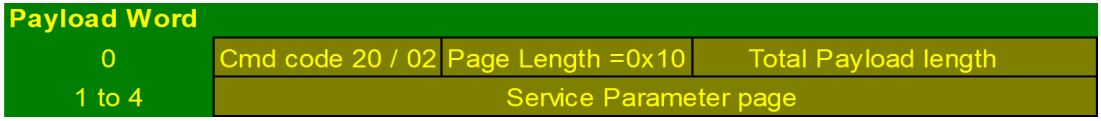


Figure 137 Process Login and Accept frame payload

Figure 137 shows the payload of the Process Login and the Accept containing the service parameters, which are in word 1 to 4 of the payload.

Frame structure in Fibre Channel

All information transferred in Fibre Channel is packaged into frames. The structure of the frame is similar to packets used in networking and consists of a frame header, data field and CRC and the frame beginning is marked with a Start of Frame delimiter and the frame end is marked with an End of Frame delimiter, shown in Figure 138.

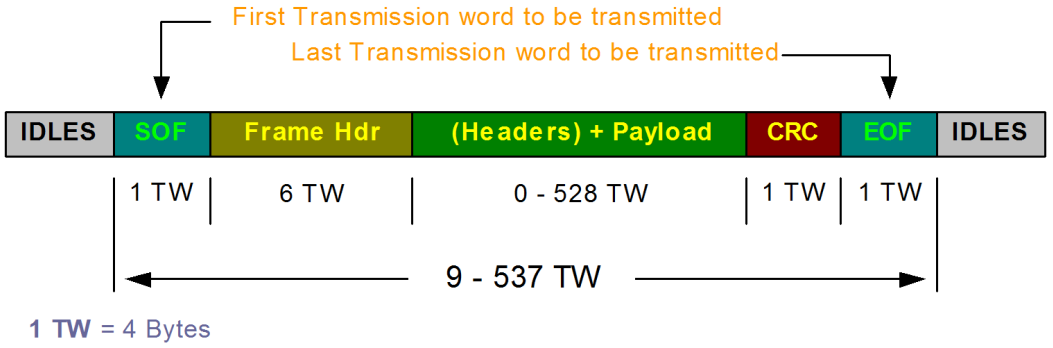


Figure 138 Frame format

The following are further discussed in this section:

- "Frame types", next
- "Start-of-frame delimiter" on page 273
- "Frame header" on page 274
- "Data/Payload" on page 281
- "Frame CRC" on page 282
- "End-of-frame delimiter" on page 282

Frame types

There are two different frame types in Fibre Channel, each of which is identified by certain bits being set in the Routing Control (R_CTL) field in the frame header. They are Frame type FT-0 and FT-1, discussed next.

Frame type FT-0

Frame type FT-0 is defined as a link control frame with a data field length of zero bytes. They are used for link control functions such as Acknowledgements, Busy, and Rejects which are used in Class 1 and Class 2 frame transmission. The R_CTL bits for this type of frame would be R_CTL= 0xCn where n=0 to 7.

Frame type FT-1

Frame type FT-1 is defined as a data frame and the data field may be any length from 0 to 2112 bytes. Examples of data frames could be frames containing data from a SCSI operation such as a READ or WRITE and also frames containing extended link data associated with the login process. Figure 139 shows a table containing information on the various frame types.

Type	Info category	R_CTL value	Link Service	Notes
Link Frame No Payload No Ack FT-0	Acknowledge (ACK)	C0-C1	Ack_0,Ack_1, Ack_N	N is the number of frames to be acknowledged
	Link Response	C2-C6	Busy F_Busy P_Busy Reject F_RJT P_RJT	 Fabric unable to deliver frame Frame has been rejected
	Link Command	C7	LCR	
Data Frame FT-1	FC-4 device data	00 - 07	SCSI,ESCON etc	Data Transfer
	FC-4 Video data	40 - 47	Reserved	
	Basic Link Service	80 - 86	Basic Link Services	
	Link Data	22 - 23	Extended Link Services Plogi,Flogi,Logo,	

Figure 139 Frame types

Start-of-frame delimiter

The start-of-frame delimiter is an ordered set that identifies the beginning of a frame. As mentioned in "Ordered sets" on page 102, additional information is also provided by the start-of-frame delimiter, such as the class of service associated with the frame, as follows:

- When the connection should be established for classes of service that establish a connection between ports.
- Whether the frame is the first frame of a sequence of frames and whether the frame is initiating the sequence or is a normal frame with a sequence.

Frame header

The frame header immediately follows the start-of-frame delimiter and is a 24-byte (6 word) structure which contains control and addressing information. It is divided into sections (Figure 140).

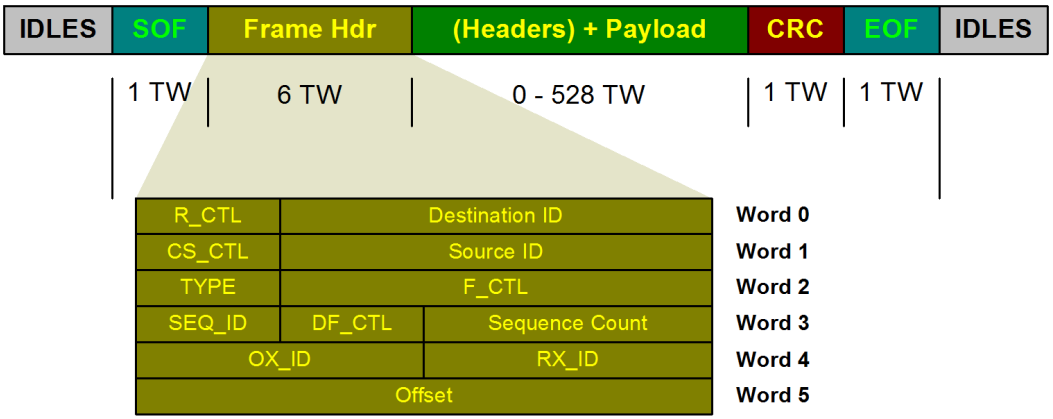


Figure 140 Frame header

The following sections shown in the frame header in Figure 140 are further described in this section:

- "Routing Control (R_CTL)", next
- "Destination ID (D_ID)" on page 275
- "Source ID (S_ID)" on page 275
- "Class Specific Control (CS_CTL)" on page 276
- "Frame Type (TYPE)" on page 276
- "Frame Control field (F_CTL)" on page 277
- "Sequence ID (SEQ_ID)" on page 279
- "Data Field Control (DF_CTL)" on page 279
- "Sequence Count (SEQ_CNT)" on page 279
- "Originator Exchange ID (OX_ID)" on page 280
- "Responder Exchange ID (RX_ID)" on page 280
- "Offset/parameter field" on page 281

Routing Control (R_CTL)

The R_CTL field (Word 0 bits 31 - 24) consists of two fields, the Routing bits and the Information field:

- The **Routing bits** (Bits 31 to 28) identify the routing of the frame (how it will be processed) and their general usage. For example if the R_CTL value is 0x0n where n=0 to 7 then you know the frame contains FC-4 device data.
- The **Information field** (Bits 27 to 24) provides further definition for that category of frame. The interpretation is dependent on the Routing bits. For example, if the R_CTL value id 0x07, then this decodes to FC_4 device data and command status. This indicates the frame payload contains command status information, which in our case this would be SCSI status. For further information on the R_CTL fields and meanings, refer to Figure 155 on page 283.

Destination ID (D_ID)

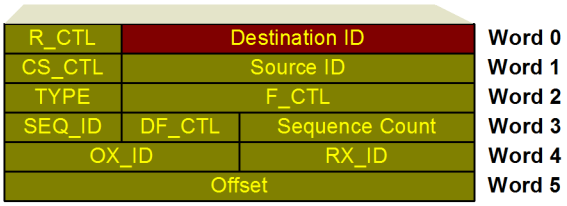


Figure 141 Destination ID field

This is a 3-byte field that contains the destination address for the frame. Every N_Port needs to have an address before it can receive frames. This address is acquired during the login process, which is different for each topology.

- In arbitrated loop, the address is acquired during the Loop Initialization stage. Only the first byte of the address is used in this topology and the other two bytes are filled with zeros.
- In fabric switch, the address is acquired during Fabric Login where the address is assigned by the fabric and all three bytes are used.
- In point-to point topology, the address is acquired during the N_Port login process.

Source ID (S_ID)

This field again is a 3-byte value containing the address of the source port of a frame. Whenever a port transmits a frame it places it address

in the S_ID field and the destinations port address in the D_ID field of the frame header.

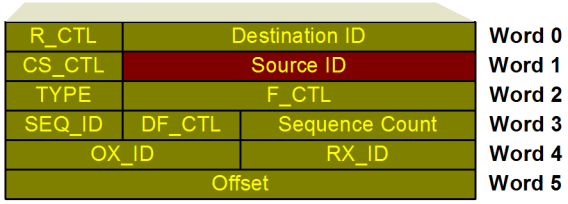


Figure 142 Source ID field

Class Specific Control (CS_CTL)

This 1-byte field is only used in Class 1 and Class 4 connections for specifying particular link speeds and link operations. This field is reserved for Class 2 and Class 3 frames.

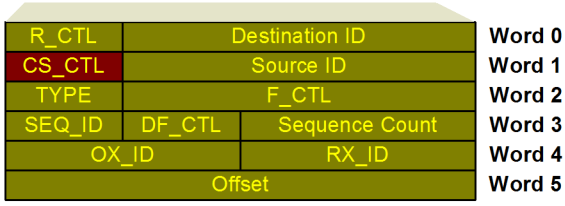


Figure 143 Class Specific Control

Frame Type (TYPE)

The Type field is a 1-byte field that is used to identify the frame depending on whether the frame is a link control frame (FT-0) or a data frame (FT-1).

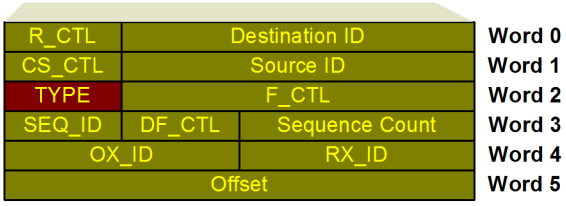


Figure 144 Frame Type field

If the frame is a link control frame (FT-0) this field is reserved except where the frame is a fabric busy (F_BSY) sent in response to a link control frame. The Type field in this case is used to indicate a reason

for the busy and reconstruct the original link control frame for retransmission.

When the frame is a data frame (FT-1) the type field is used to identify the protocol being carried by the frame. For example, a type field of "08" indicates SCSI FCP.

For more information on these frame types, refer to "Frame types" on page 272.

Frame Control field (F_CTL)

This is a 3-byte field that contains control information relating to the frame content, as shown in Figure 145. This portion of the frame header is used to identify Exchange control, Sequence control, Acknowledgement policy, and Abort Sequence actions.

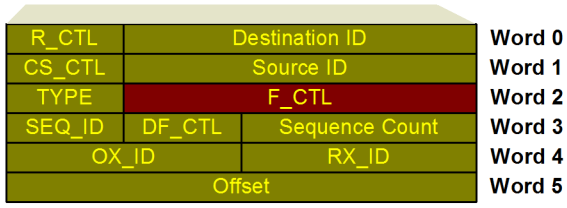


Figure 145 Frame Control field

A breakdown of all the bits in the F_CTL field is shown in Figure 146, which continues on the next page.

F_CTL Bits	Control field	Description
23	Exchange_Context	0 = Originator of Exchange 1 = Responder of Exchange
22	Sequence_Context	0 = Sequence Initiator 1 = Sequence Recipient
21	First_Sequence	0 = Not the first sequence of the exchange 1 = First sequence of the exchange
20	Last_Sequence	0 = Not the last sequence of the exchange 1 = Last sequence of the exchange
19	End_Sequence	0 = Not the last data frame of the sequence 1 = Last data frameof the sequence
18	End_Connection Class 1	0 = Connection active 1 = End class 1 connection pending
	Deactivate_Class_4_Circuit	0 = Class 4 circuit active 1 = Deactivate class 4 circuit

Figure 146 Frame Control (F_CTL) field

F_CTL Bits	Control field	Description
17	Chained_Sequence Reserved	0 = No Chained_Sequence 1 = Chained-Sequence active
16	Sequence_Initiative	0 = Sequence Initiative held 1 = Sequence Initiative transferred
15	X_ID reassigned	0 = X_ID assignment retained 1 = X_ID reassigned
14	Invalid X_ID	0 = X_ID assignment retained 1 = Invalid X_ID
13 ,12	Ack_Form	00 = No assistance provided 01 = Ack_1 required 10 = Ack_n required 11 = Ack_0 required
11	Data Compression (reserved)	0 = Uncompressed frame payload 1 = Compressed frame payload
10	Data Encryption (reserved)	0 = Unencrypted frame payload 1 = Encrypted frame payload
9	Retransmitted_Sequence	0 = Original sequence transmission 1 = Sequence retransmission
8	Unidirectional Transmit Class1	0 = Bidirectional transmission 1 = Unidirectional transmission
	Remove Class 4 circuit (Class 4)	0 = Retain or deactivate class 4 circuit 1 = Remove class 4 circuit
7,6	Continue Sequence Condition	Last data frame - sequence initiator 00 = next sequence to follow -no information 01 = next sequence to follow - immediately 10 = next sequence to follow soon 11 = next sequence to follow delayed
5,4	Abort Sequence Condition	In an ACK or RJT frame from sequence recep 00 = Continue sequence 01 = Abort sequence ,perform ABTS 10 = Stop sequence 11 = Immediate sequence retransmission requ

Figure 146 (continued) Frame Control (F_CTL) field

Sequence ID (SEQ_ID)

The Sequence ID field is a 1-byte field used to track all frames in a particular sequence between two ports. All frames in the sequence will have the same SEQ_ID value in the frame header. This value is assigned by the sequence initiator and must be unique between the two ports while the sequence is open.

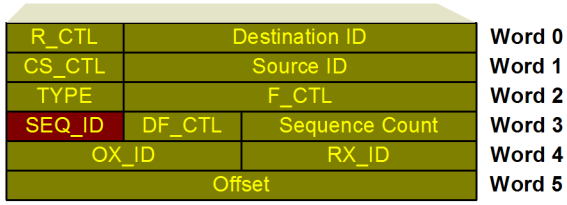


Figure 147 Sequence ID field

Data Field Control (DF_CTL)

The Data Field Control is a 1-byte field that defines whether optional headers will exist at the beginning of the data field of the frame. These optional headers are used by some applications and protocols but are not present in all frames. They basically are a means of extending the frame header into the data field for applications that require additional header information. For further information, please see Figure 155 on page 283.

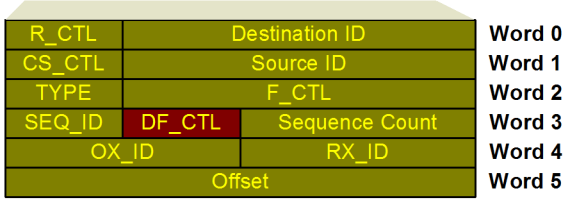


Figure 148 Data Field Control field

Sequence Count (SEQ_CNT)

The Sequence Count field is a 2-byte field that indicates the sequential order of a frame within a sequence. The SEQ_CNT value is included in every frame header and is incremented by one from the previous frame in the sequence.

This field is used in determining the order frames are received, that all frames in a sequence have been received and to detect lost or missing frames.

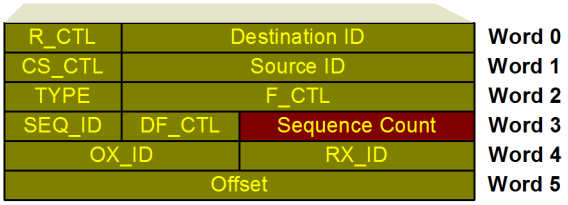


Figure 149 Sequence Count field

Originator Exchange ID (OX_ID)

The Originator Exchange ID is a 2-byte field used to identify all frames that are part of an exchange. Once an OX_ID is assigned to an exchange it is used in every frame of the exchange whether the frame is sent by either the originator or the responder.

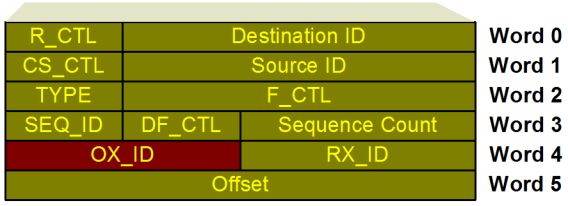


Figure 150 Originator Exchange ID field

Responder Exchange ID (RX_ID)

The Responder Exchange ID is also a 2-byte field used to identify frames that are part of a specific exchange. This value is assigned by the exchange responder and is put in every frame sent by the responder, which in most cases will be the Symmetrix FA and is also present in every frame echoed back by the exchange originator.

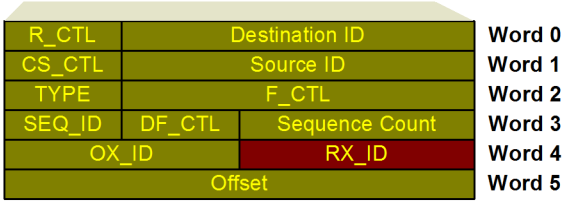


Figure 151 Responder Exchange ID field

Offset/parameter field

This 4-byte field in the frame header is used for different things dependent on the frame type.

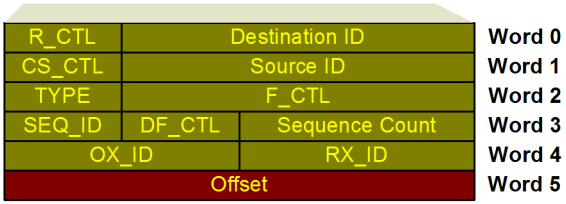


Figure 152 **Offset/parameter field**

If the frame is a Link Control frame such as an Acknowledge (ACK) the parameter field indicates the number of frames to be acknowledged. This field is also used to provide information for Port Busy and Port and Fabric rejects.

If the frame is a data frame, then the parameter field contains a value indicating the offset of the first byte of data in the payload relative to the ULP defined buffer for that I/O operation. The existence of a relative offset is indicated by bit 3 in the F_CTL field. The relative offset is used in reassembly of sequences at the receiver end as it provides an offset as to where the payload of each frame needs to be placed in the ULP buffer to reassemble sequences/information units.

Data/Payload

This part of the Fibre Channel frame is a variable length field and can contain optional header information (as mentioned in the DF_CTL field), SCSI data and in certain cases fill words. The maximum size of the data field a frame is 2112 bytes. The contents of the frame payload is determined by the FC-4 layer. In our case this will be SCSI information such as command, data or status information being transported between a source (HBA) and destination N_Port (Symmetrix FA). Refer to Figure 153 on page 282.

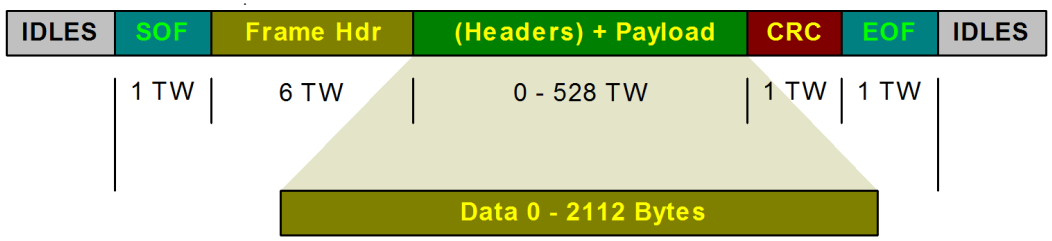


Figure 153 Payload

Frame CRC

This 4-byte value is used to ensure that the contents of the frame header and data field information in the frame have been received correctly. The CRC field follows the data field and is before the EOF delimiter. This value is calculated prior to encoding for transmission and after decoding at reception. SOF delimiters and EOF delimiters are not used in the calculation of the CRC value.

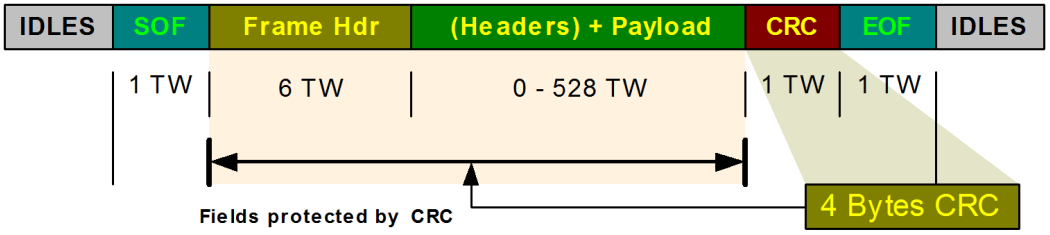


Figure 154 CRC protection

End-of-frame delimiter

The end-of-frame delimiter is an ordered set that identifies the end of a frame to the receiving port. It also removes the connection for the classes of service that establish a connection such as class 1 and also signifies whether the frame is the last frame in a sequence or a normal frame within a sequence.

Figure 155 on page 283 further defines the information contained in frames.

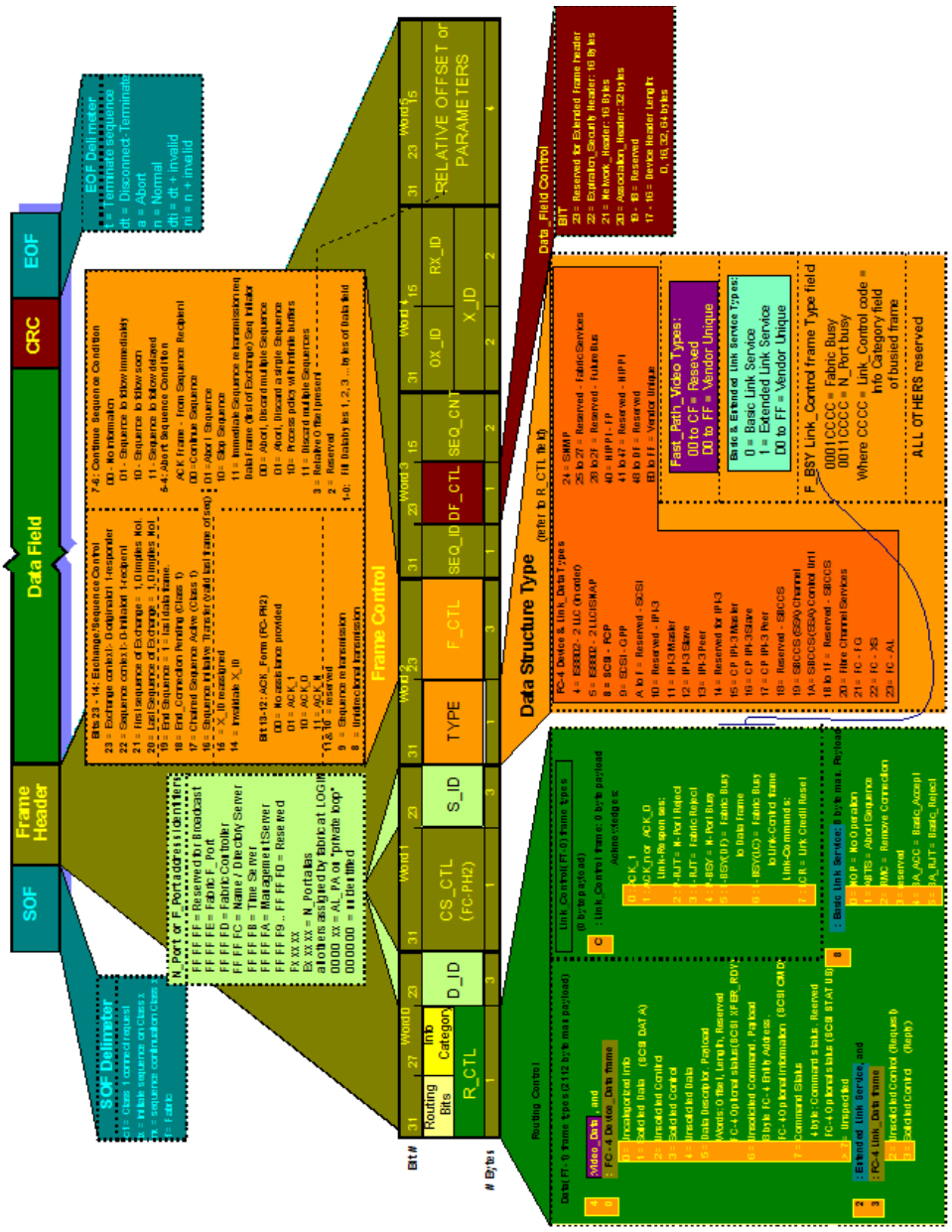


Figure 155 Frame definition

Class of Service (C.O.S.)

Fibre Channel defines different delivery options for frame transmission and these are known as a particular Class of Service. Class 1, 2, and 3 are the different services supported by Fibre Channel and are basically defined with regards to connection, in-order delivery, confirmation of delivery, and which type of flow control is used. These classes of service are independent of the topology used.

Class 1

Class 1 is a connection-oriented service that provides a dedicated connection between two ports allowing them to use the maximum available bandwidth. Class 1 provides confirmation of delivery and notification of non-delivery. As shown in Figure 156 on page 285, a connection is established at the start of the sequence with the SOF(C1) delimiter and the return of the ACK from the recipient. BB_Credit is only used on the request frame and the ACK. Once the connection is established, frame flow is controlled with End-to-End credit.

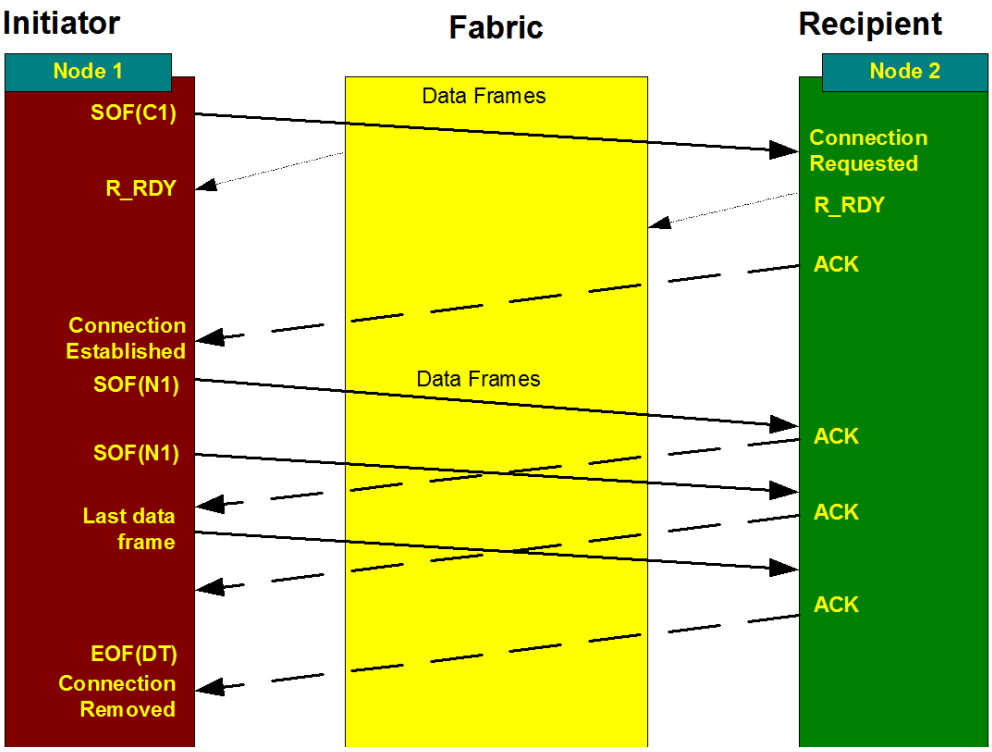


Figure 156 Class 1 Dedicated connection

When the connection is no longer needed the port that is the sequence initiator at the time sets the End_Connection bit in the F_CTL of the last data frame and the recipient replies with an ACK that has the end-of-frame disconnect terminate EOF(DT) delimiter set.

This removes the Class 1 connection. Class 1 thus provides guaranteed bandwidth, in-order delivery, and confirmation of delivery.

Class 2

Class 2 is a connectionless class of service that provides confirmation of delivery and non-delivery of frames. No connection is established between the source and destination ports and it is up to the fabric to deliver the frames, possibly over different paths to the recipient. Frames will be transmitted in order but depending on the fabric may

arrive at the recipient out of order. Frames may be sent to different destinations one after the other and it is possible to be transmitting a frame to one destination while receiving a frame from another which means Class 2 multiplexes on a frame-by-frame basis.

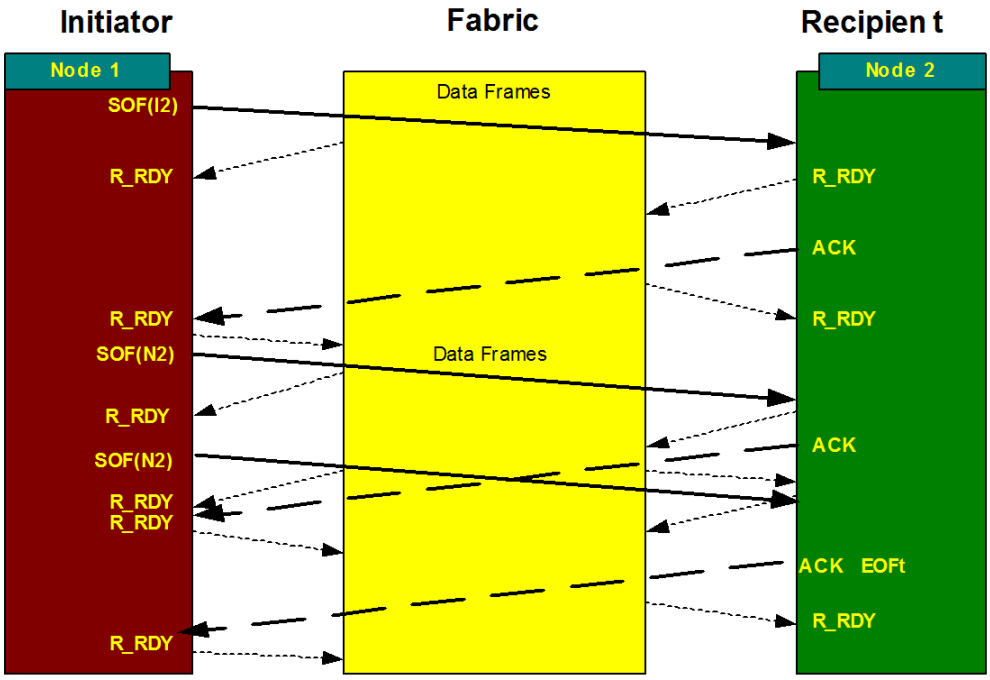


Figure 157 Class 2 operation

Figure 157 shows the Buffer-to-Buffer (R_RDY) and End-to-End (ACK) flow control in Class 2 frame flow. The sequence is started with a Start of Frame Initiate Class 2 SOF (I2) delimiter and there is no disconnect as there is in Class 1 but the last frame in the sequence uses a End of Frame terminate (EOFt) delimiter. The initiator sets the End_Sequence in the F_CTL field of the frame header of the last data frame and the recipient sets the End_Sequence and the EOFt delimiter on the last ACK of the sequence.

Class 3

Class 3 is again a connectionless class of service but with no confirmation of delivery or non-delivery of frames. No connection is established and, like Class 2 frames, Class 3 frames may arrive out of order. The biggest difference is that the delivery of frames is unacknowledged in that the destination port does not send any link control frame (ACK) on receipt of valid data frames. The flow control used in Class 3 is Buffer-to-Buffer, which uses the R_RDY primitive signal.

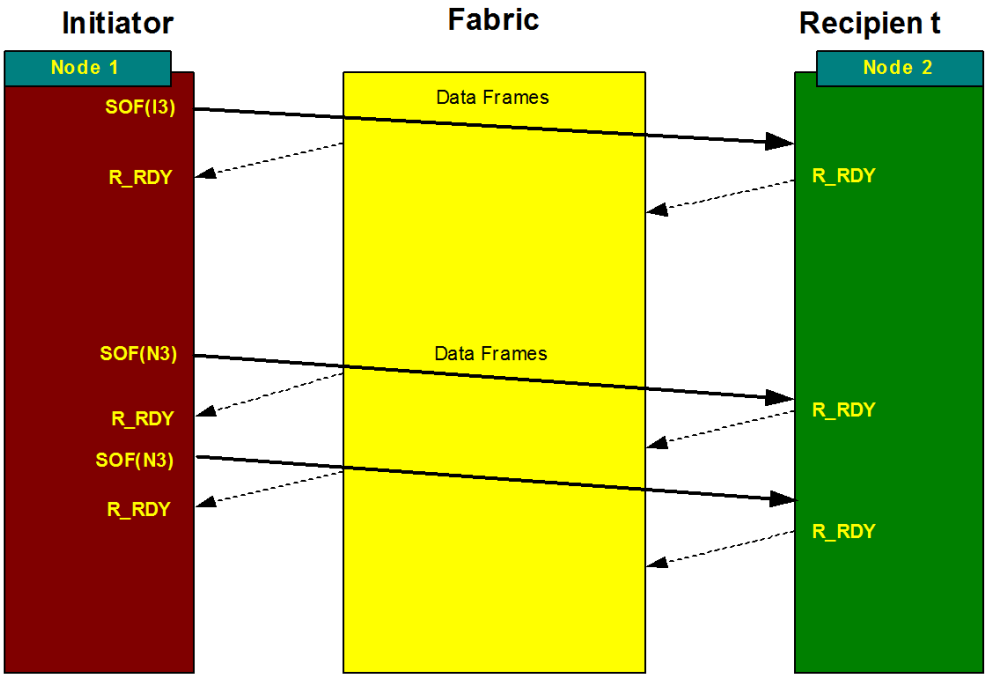


Figure 158 Class 3 Flow control

The fact that there is no confirmation of delivery of frames in Class 3 Fibre Channel means error recovery must be handled at the Upper Layer Protocol. The confirmation is provided by the protocol being transported, which in our case will be SCSI.

Class 3 is the class of service used by Symmetrix and as SCSI itself is an acknowledged protocol it can handle error detection and recovery.

Figure 159 shows a class of service summary.

	Class 1	Class 2	Class 3
Connection type	Dedicated Connection	Connectionless	Connectionless
Flow Control	End-to-End Credit	End-to-End Credit Buffer -to-Buffer	Buffer-to-Buffer credit
Frame delivery	In order delivery	Order not guaranteed	Order not guaranteed
Frame acknowledged	Acknowledged	Acknowledged	Unacknowledged
Route used	Circuit established / removed	Frame by Frame routing	Frame by Frame routing
Multiplexing	No	Yes	Yes
Bandwidth	Poor bandwidth utilization	High bandwidth utilization	High bandwidth utilization

Figure 159 Class of service summary

Buffer-to-buffer credit (BB_Credit)

Fibre Channel uses the BB_Credit (buffer-to-buffer credit) mechanism for hardware-based flow control. This means that a port has the ability to pace the frame flow into its processing buffers. This mechanism eliminates the need of switching hardware to discard frames due to high congestion. EMC testing has shown this mechanism to be extremely effective in its speed and robustness.

For more information on BB_Credit, refer to the "Data buffering and flow control" section in the *Extended Distance Technologies TechBook*, l located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab. For information on buffer-to-buffer flow control, refer to "Buffer-to-buffer flow control" on page 221.

Zoning

Zoning is the process of grouping initiator and targets into zones. Initiators and targets placed in the same zone are allowed to communicate by the fabric. Zoning also plays a critical role in event isolation, i.e., RSCN (Registered State Change Notifications) distribution. In order for a fabric to work properly, it *must* be properly zoned. Currently, a few types of zoning are supported, each discussed briefly below:

- World Wide Port Name (WWPN) zoning

  By far, the most common form of zoning is by WWPN. You select the unique 64-bit addresses of the initiator, its target(s), and place them in a common zone. Its advantage is that no matter where you attach the WWPN, as long as it is in the same fabric, it will always be able to discover, and be discovered, by other ports to which it has been zoned to have access.

- Domain, Port (D,P) zoning

  This is another common form of zoning. You specify an initiator and target by their physical location in the fabric. The main advantage is perceived to be security, and in some cases it is more secure. Refer to the *Building Secure SANs TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab, for more information.

- World Wide Node Name (WWNN) zoning

  Although it has not been used much up to this point, there is potential for it to be used extensively in clustered environments.

Storage

There are many types of storage available including tape, disk, and WORM (Write Once Read Many). There are numerous ways to implement and present the storage to the end user. A tape, for example, can be presented directly as a tape drive on the SAN or it can be front-ended with a disk-based library, such as CDL or SDL. The same is true for disk-based storage. It can be presented as a physical disk, as is the case with a JBOD (Just a Bunch Of Disks) solution, or as a virtual disk, as is the case with most array-based storage solutions. This section provides a high-level overview of the major storage concepts, as follows:

- Disk
  A disk is the physical media that data will be stored on. Disks are typically carved up or concatenated to make logical volumes of the desired size.
- Volume (aka Logical Volume)
  A volume is an entity that can be made available for use on a front-end port at a specific LUN address. It can also be thought of as a virtual disk.
- LUN (Logical Unit Number)
  A LUN is the address at which a volume is presented to the user on a front-end port.
- Front-end port
  A front-end port is an interface, either FC or IP, on which LUNs are presented for customer use.
- Array controller
  An array controller is the collection of hardware and software features used to convert physical disks into addressable volumes. The array controller also provides a cache for temporarily storing data while it is waiting to be written to disk.
- Cache
  Cache is a high bandwidth/low response time temporary data storage area provided by the array controller.
- Back-end port
  A back-end port is an interface, either FC or IP, that the array controller uses to push data to the disk.

NPIV

The following information is discussed in this section:

Overview

Traditional N_Port initialization works as follows (refer to Figure 160 on page 293):

- N_Port sends FLOGI to Fabric Server (0xFFFFFE) to obtain a valid address.
- Fabric Server responds with an Accept containing the assigned address.
- N_Port sends PLOGI to the Name Server (0xFFFFFC) to start the registration process.
- N_Port sends SCR to the Fabric Controller (0xFFFFFD) to register for state change notifications (RSCNs).

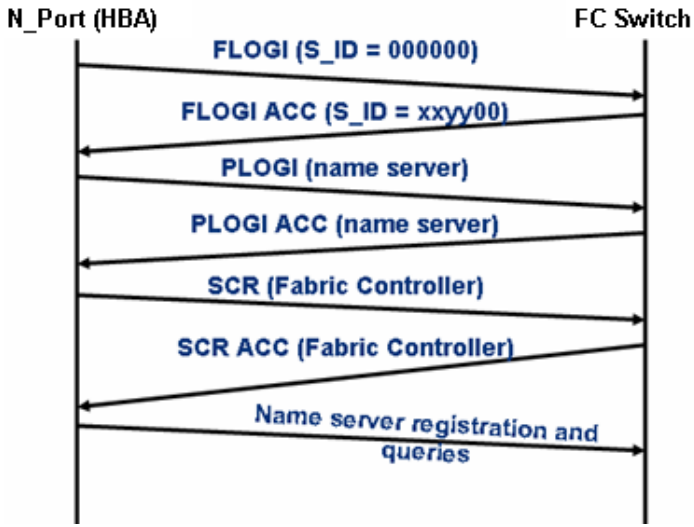


Figure 160 Traditional N_Port initialization

NPIV-capable N_Port initialization works as follows (refer to Figure 161 on page 294):

- N_Port sends FLOGI to Fabric Server (0xFFFFFE) to obtain a valid address.
- Fabric Server responds with Accept with the assigned address.
- N_Port sends FDISC to Fabric Server (0xFFFFFE) to obtain a virtual address.
- Fabric Server responds with Accept with the assigned virtual address N_Port.
- To acquire an additional address for NPIV session, repeat FDISC as additional NPIV sessions as needed.
- Each virtual ID has its own WWN.
- Virtual IDs can be zoned as any other normal N_Port.
- Virtual IDs can register and query the name server.

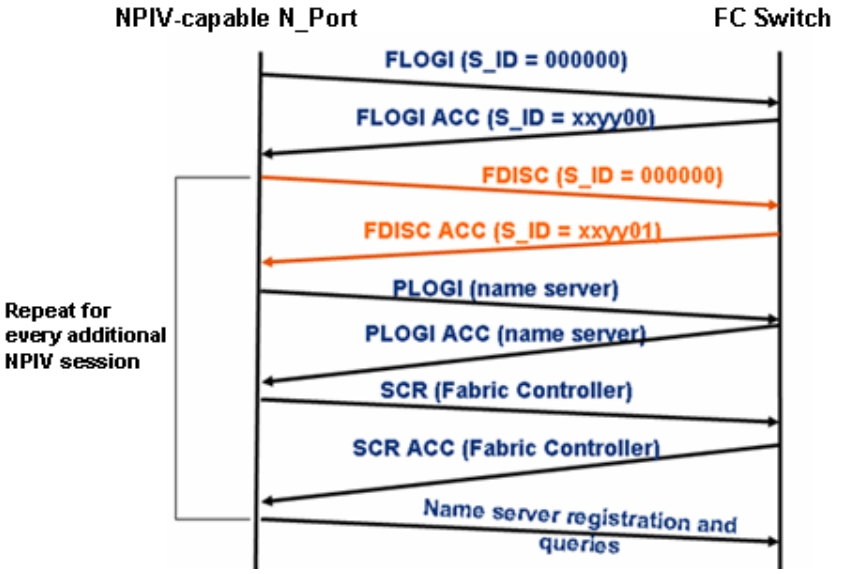


Figure 161 NPIV-capable N_Port initialization

NPIV is useful in environments with Blade Servers, Multi ID devices (tape, encryption) and Server Virtualization, each discussed in more detail in this section.

Blade servers

Blade servers that utilize Fibre Channel Switch Modules present a challenge to SAN scalability and interoperability as each chassis contains two FC switch modules (domains), as shown in Figure 162.

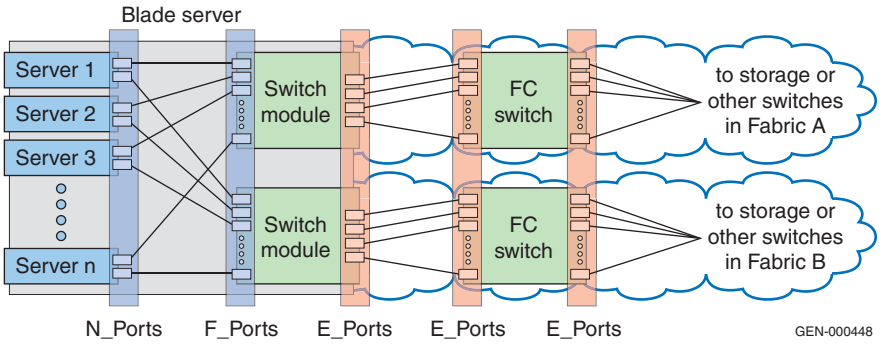


Figure 162 Blade servers

It is important to stress that each of the FC switch modules embedded in the blade server is a full-blown FC switch with a name server, Domain ID, etc. Because of this, due to FC switch interoperability concerns, each blade server vendor needs to provide a switch module capable of connecting to a SAN consisting of switches from each of the switch vendors. For example, if a blade server were going to be attached to a Brocade fabric, a Brocade switch module would be the best choice to embed in the blade server, and not a QLogic module.

A drawback to having two switch modules in each chassis is that two domain IDs are consumed for each chassis. Since most vendors only support between 31 and 50 Domain IDs, this can present a scalability problem. In fact, depending on how many blade servers are added, this can easily constrain the total fabric size to less than 800 ports, which is far short of the minimum number of supported N_Ports in today's fabrics of 2048.

Another consideration is that by embedding the FC switch into the Blade Server, you are also choosing to embed higher risk interoperability points into the server chassis. The E_Ports are a higher risk due to the nature of FC-SW interop (see EMC Knowledgebase article EMC149735 for more details on current FC-SW interop problems).

To eliminate these problems, the switch modules can be replaced with NPIV gateways (Figure 163). This improves fabric scalability and interoperability since each NPIV gateway presents only additional N_Ports to the fabric and not E_Ports or additional domains.

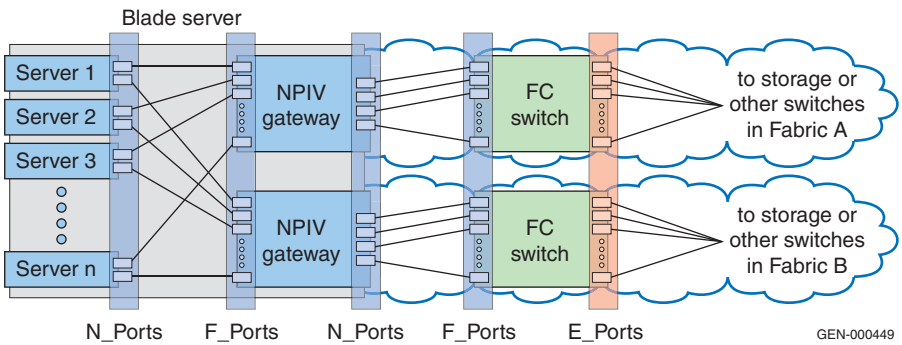


Figure 163 NPIV gateways

Another advantage is that the higher risk interoperability points are moved out of server chassis. Finally, server blade HBAs use standard drivers and this, in combination with the NPIV gateway's level of transparency, should allow for a higher level of interoperability.

Multi ID devices

Another application for NPIV is as a replacement for the arbitrated loop functionality used by some multi ID devices, such as the CLARiiON Disk Library or Decru DataFort. These devices use public loops to present multiple IDs into the fabric. Outside of the operational inefficiencies of using a loop during normal operation, the largest drawback to loops is that the addition of a new ID (as happens during failover on dual-headed CDL configurations) requires a reset of the loop and is very disruptive to I/O. Another drawback is that it requires loop support on the FC switch.

Figure 164 through Figure 167 on page 298 highlight the difference between failover on CDL when Arbitrated loop is used versus NPIV.

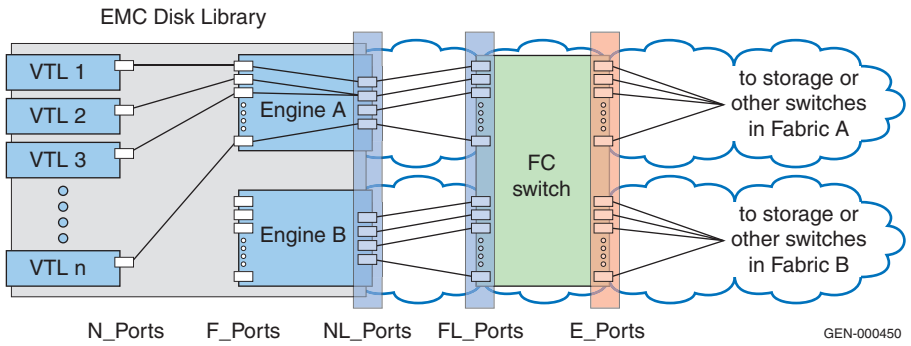


Figure 164 Normal operation with FC-AL

In Figure 165, Engine A fails.

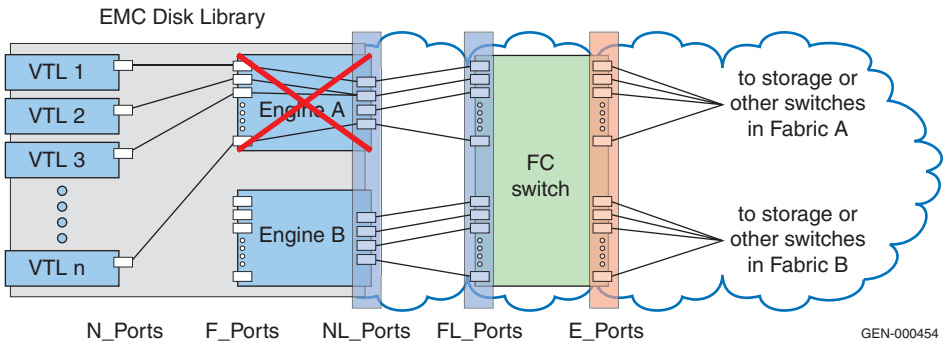


Figure 165 Engine A failure

Failover to Engine B results in LIP generation and disruption to established I/O (Figure 166).

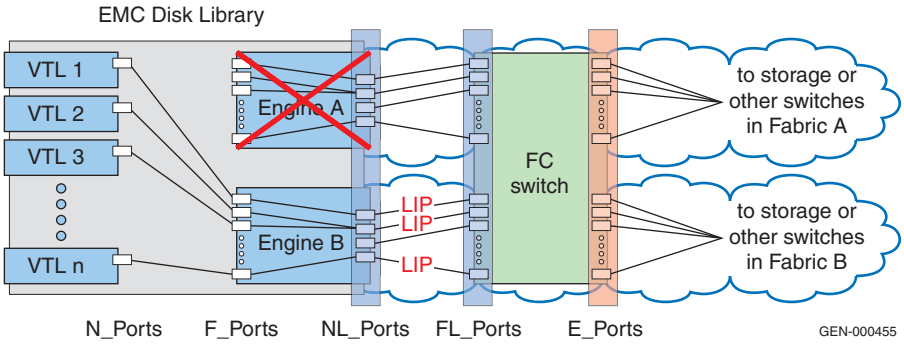


Figure 166 Failover to Engine B

When NPIV is used under the same type of failure, FDISC are sent instead of LIP. The transmission of FDISC does not impact existing I/O (Figure 167).

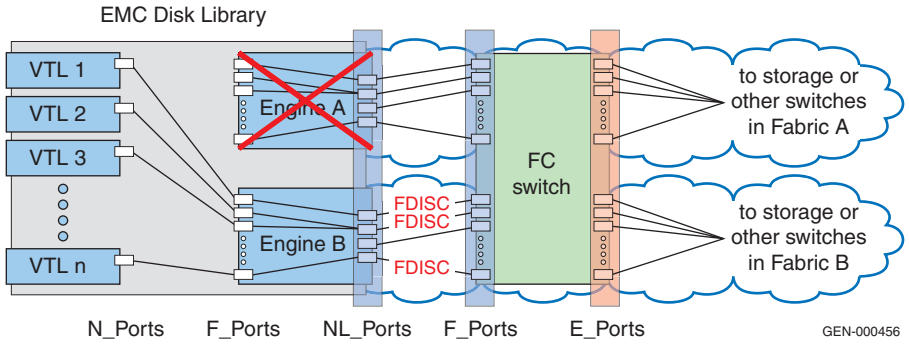


Figure 167 Using NPIV

Server virtualization

NPIV can also be used in conjunction with server virtualization. Today this is limited to mainframe z/Linux environments, but this may eventually be expanded to include support for VMware. The idea is that each virtual machine have its own WWN. This is useful when SAN Management is an entirely separate discipline and organization from Server Management because existing SAN Management tools can be used for storage provisioning.

NPIV challenges

NPIV has the potential to simplify the SAN, but it does have some trade-offs, the primary one being that there are not many implementations utilizing NPIV today. In spite of the idea having been around for a while, the adoption has been slow until at least early 2007. It is suspected that with the release of NPIV gateways, NPIV's popularity will increase or decrease in proportion to the number of new blade server deployments.

Another important consideration is that even though there are no FC-SW interop concerns, this does not mean there are no interoperability concerns. Access Gateway from Brocade, along with similar implementations, still presents devices as N_Ports so there are

inherent bandwidth sharing concerns that will have an unknown impact.

Finally, once NPIV has been deployed in your environment, you will have eliminated the domain count restriction, but the maximum number of N_Ports supported in a single fabric (2048-4096) may come soon, so scalability concerns are deferred, but not resolved. To resolve the scalability concerns you will need to deploy Fibre Channel routers.

Fibre Channel Routing

Fibre Channel Routing is a Fibre Channel facility that allows ports to be shared across separate SANs without merging them. It is defined in FC-IFR (Fibre Channel Inter-Fabric Routing), which is a Fibre Channel standard in *working-draft proposed* status. The purpose of FC-IFR is to reduce the scalability limits imposed by the name server by only sharing certain resources between fabrics. By not collapsing all of the N_Ports into one large name space, Domain ID limits are avoided, as are the current maximum number of N_Port limitations. It provides solutions for configurations with FC-SW interoperability, resource consolidation, distance extension, or scalability concerns, each discussed further in this section.

This section discusses the following:

As shown in Figure 168 on page 301, routing spans many different topologies and protocols. The simplest concept is FC-to-FC routing where two fabrics connect to the same router. A bit more complicated is connectivity over two FC routers, which are connected by an IP link or a backbone fabric.

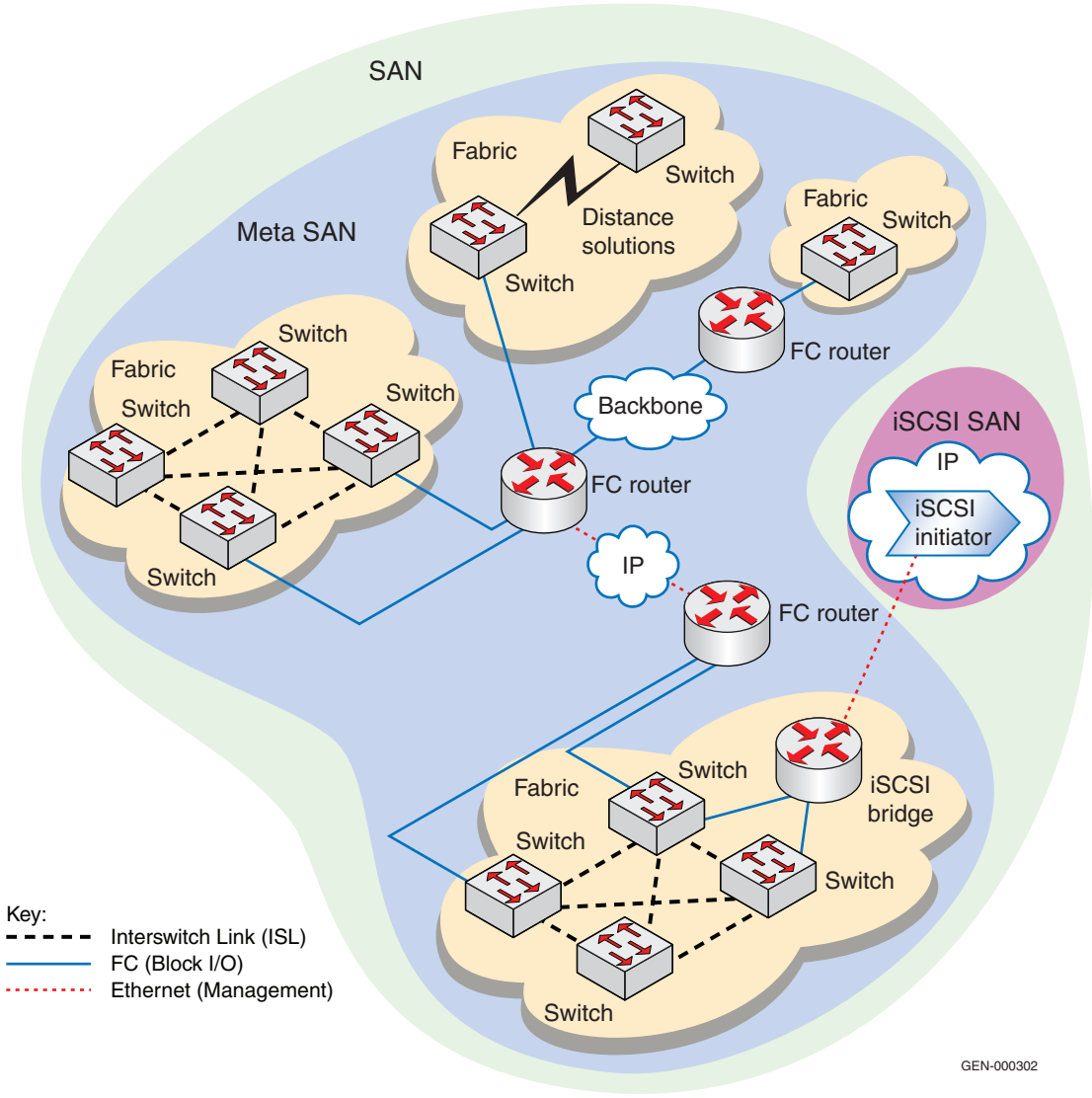


Figure 168 Fibre Channel routing

Interoperability

Figure 169 shows four different fabrics, none of which are compatible with each other due to Domain ID overlap, different operating modes, and even different firmware revisions.

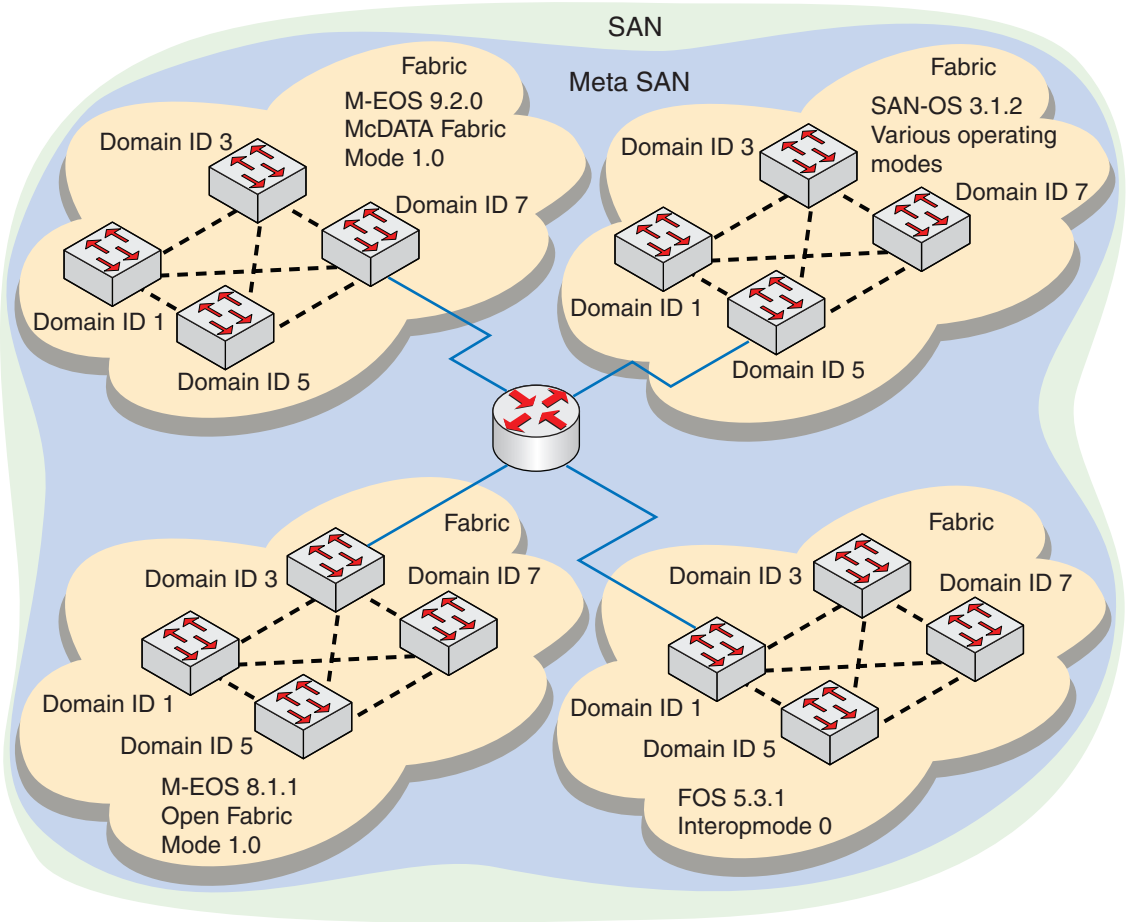


Figure 169 Interoperability in Fibre Channel routing

Sometimes, due to uptime requirements, it is not possible to take a fabric down to change Domain IDs or operating modes, or to upgrade switch firmware because of interoperability concerns

between older, but established, OS/HBA/driver and storage combinations.

It is also possible that you will be required to establish connectivity between devices that reside on each of these fabrics. In this case, you will need to use a router to act as a bridge between the different fabrics.

Resource consolidation

A specific example of why interoperability between different fabrics would be desirable is an environment that needs to deploy tape backup. In Figure 170 on page 304, tape (i.e., CDL) is connected to the McDATA fabric while two of the servers that need to access it are connected to other fabrics with different operating modes and overlapping domain IDs. You may be able to just attach the backup servers and the CDL to the same fabric as long as the storage that needs to be backed up can also be moved to the new fabric. Odds are, though, that this type of a move will *not* be possible in your environment.

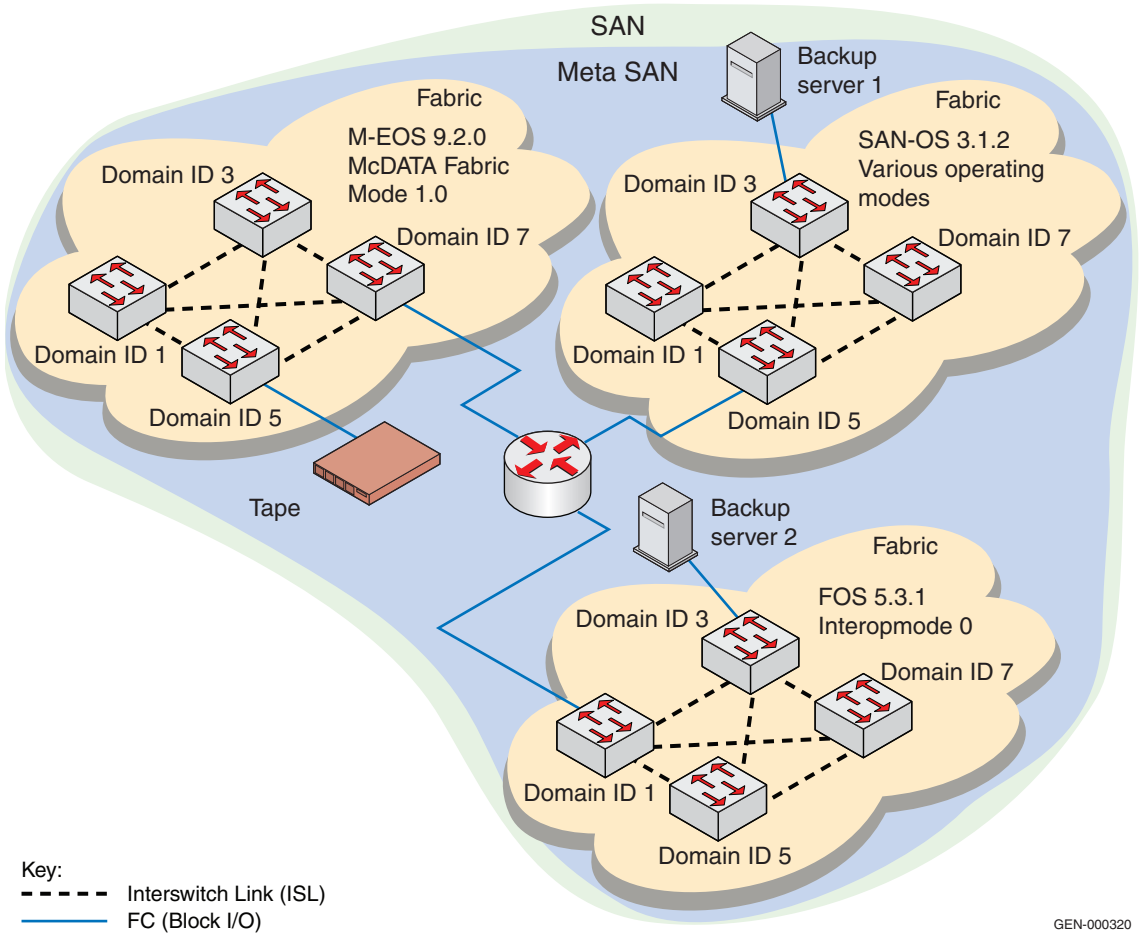


Figure 170 Resource consolidation in Fibre Channel routing

Distance extension

Another example of where routing is beneficial is when you have a need for a distance extension solution (Figure 171).

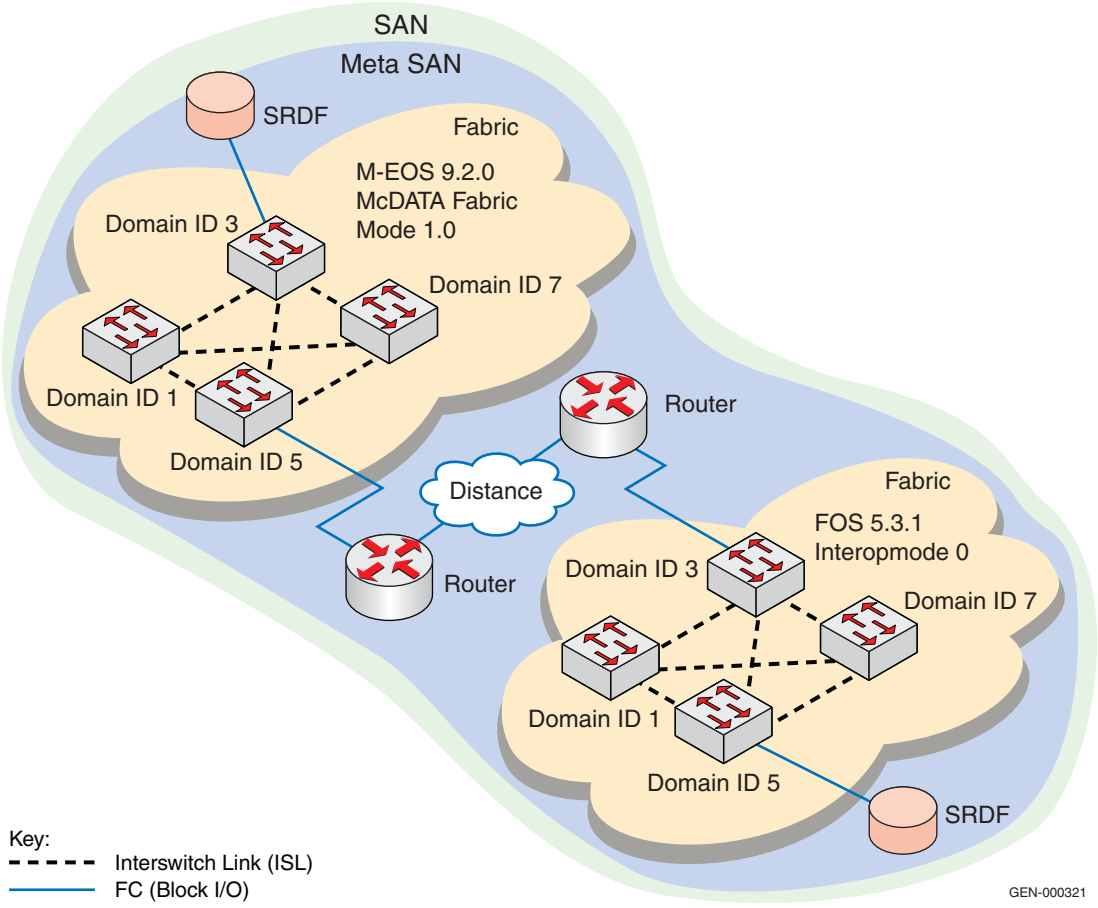


Figure 171 Distance extension in Fibre Channel routing

There are many solutions available, such as FCIP and DWDM, that will enable connectivity over long distances for the purposes of supporting SRDF or MirrorView. The majority of these solutions will simply extend the ISL and allow Fibre Channel to handle buffering and error recovery. When using native Fibre Channel for long distance links, the performance of the link is limited both by the efficiency of the application using the link and the number of

buffer-to-buffer credits available. Introducing FastWrite and increasing the buffer count can overcome both of these limitations.

Error recovery, however, cannot be handled so easily. When ISLs are added to or removed from a fabric, especially ones that connect two fabrics over distance, performance throughout the entire fabric can be *negatively* impacted. To minimize the impact that long distance link disruptions have on the fabric, routers can be deployed at each edge of the fabric to ensure that each fabric is isolated from these temporary disruptions.

Scalability

As discussed in "Maximum number of Nx_Ports in the fabric" on page 76, there is a limit to the number of ports that a single fabric can support. With today's firmware revisions, this number is in the range of 2048-4096 ports, depending upon the platform. Figure 172 on page 307 is intended to illustrate many large fabrics connected by a backbone fabric. Configurations of this type have been successfully deployed by several customers.

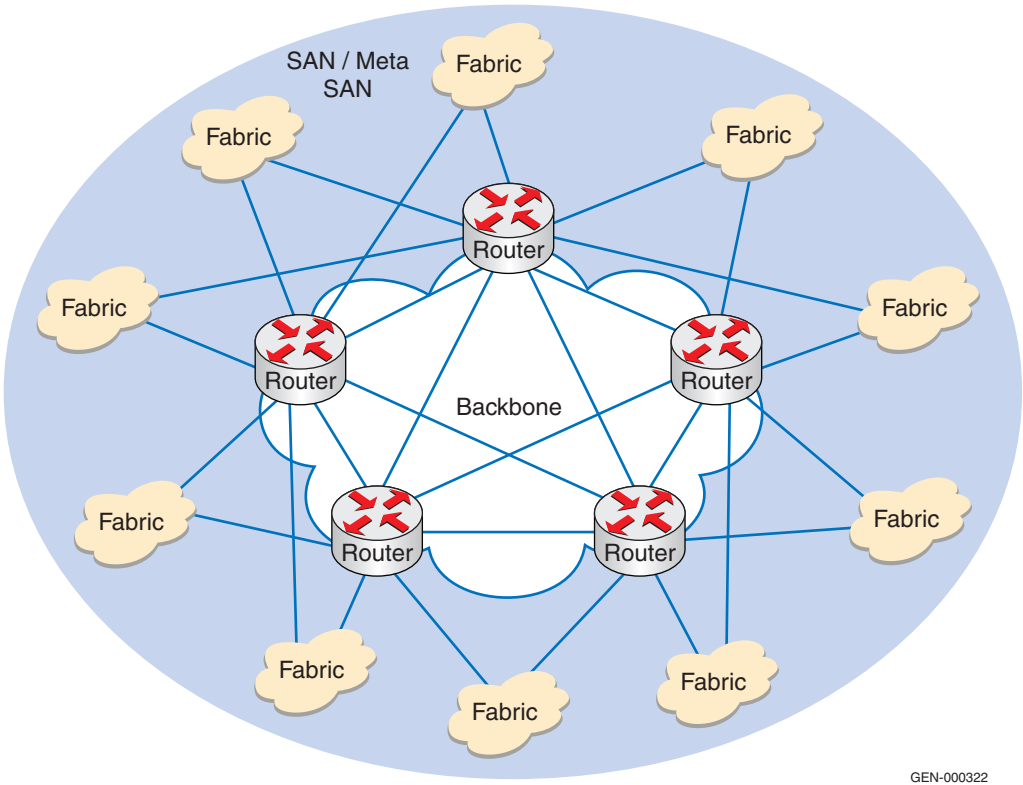


Figure 172 **Scalability in Fibre Channel routing**

The advantage to this configuration is that individual devices can be shared between fabrics without running into scalability limits.

Limitations

Consider the following limitations when using routing:

Routing is an excellent solution if you are encountering compatibility issues or experiencing scalability limits. However, mostly due to a lack of good management tools, it adds a great deal of complexity to a fabric. This causes the fabric to be difficult to service and troubleshoot, especially when dealing with end-to-end connectivity problems and Network Address Translation is being used.

Another consideration is manageability. A routed environment is more difficult to manage. Simple tasks, such as zoning, are more

difficult to accomplish when each fabric connecting to a router needs to have its zone manually updated. New products are being developed, such as Connectrix Manager that is capable of automatically activating the zone set on the required fabrics, but this product currently only works on the MP-2640M and MP-1620M models.

Although scalability limitations are minimized, connectivity is minimized as well. Routing does not provide any-to-any connectivity. This limitation will be masked somewhat once the management tools are enhanced, but it is important to realize that there are limitations to the number of ports that can be shared between any two fabrics.

Brocade SAN Routing — FCR

The Fibre Channel routing service allows nodes in two or more separate fabrics to communicate without merging those fabrics. The Fibre Channel routing service can be simultaneously used as a SAN router and for SAN extension over wide area networks (WANs) using FCIP.

Fibre Channel routing (SAN routing and SAN extension) supports interoperability with Connectrix M Series fabrics (either in McDATA mode (native mode) or Open Fabric mode).

Note: The SAN router MP-7500B and SAN router FR4-18i blade are referred to as a "SAN router" throughout this chapter.

Note: In this chapter, the term "MetaSAN" is used for the collection of all SANs interconnected with Fibre Channel routers (see "MetaSAN" on page 311).

SAN routing concepts

SAN routing introduces the following concepts:

Logical Storage Area Networks (LSANs)

An LSAN is defined by specially named zones (LSAN zones) in two or more edge or backbone fabrics that contain the same device WWPNs. These LSAN zones enable Fibre Channel zones to cross physical SAN boundaries without merging the fabrics while

maintaining the access controls of zones (see Figure 173).

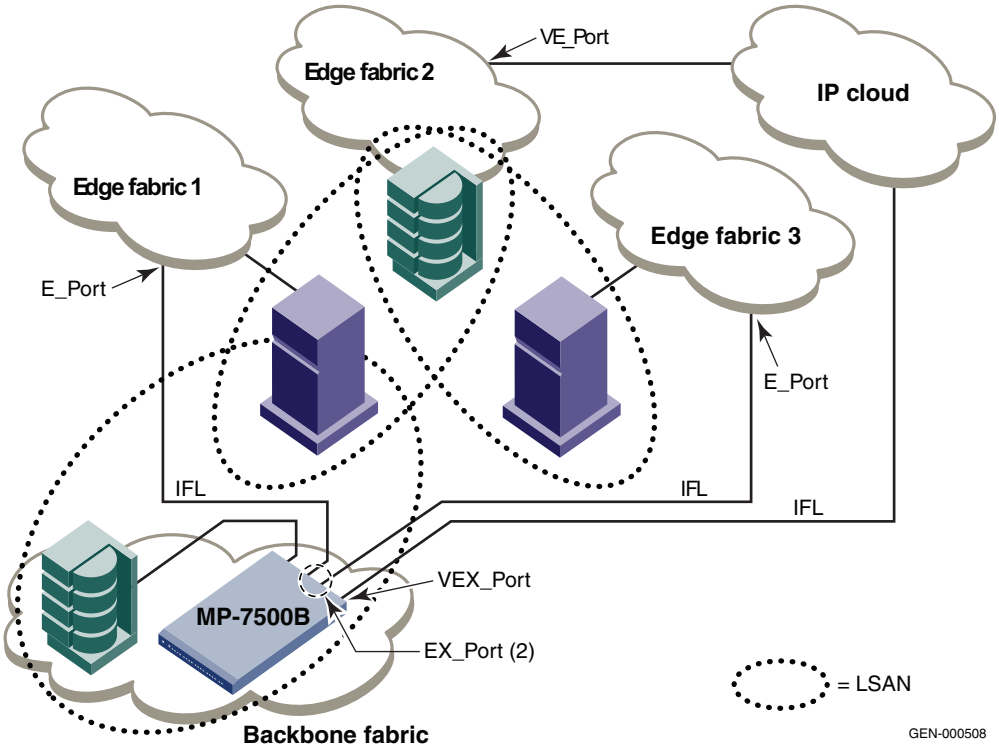


Figure 173 MetaSAN with edge-to-edge and backbone fabrics

Figure 173 shows a MetaSAN with a backbone consisting of one SAN router (Connectrix MP-7500B) connecting hosts in Edge Fabric 1 and 3 with storage in Edge Fabric 2 and the backbone through the use of LSANs. There are devices shared between the backbone and Edge Fabric 1, between Edge Fabric 1 and Edge Fabric 2, and between Edge Fabric 2 and Edge Fabric 3.

EX_Port

A special type of port, called an EX_Port, functions similar to an E_Port, but terminates at the switch and does not propagate fabric services or routing topology information from one edge fabric to another. The link between an E_Port and an EX_Port is called an *interfabric link* (IFL). You can configure multiple IFLs from a SAN router MP-7500B and a Connectrix ED-48000B with the SAN router FR4-18i blade using chassis configuration option 5.

Note: The SAN router MP-7500B and SAN router FR4-18i blade are referred to as a "SAN router" throughout this chapter.

Fibre Channel routing services support multiple EX_Ports connected to the same edge fabric. Figure 174 shows a MetaSAN consisting of three edge fabrics connected through a Connectrix ED-48000B containing an FR4-18i with interfabric links.

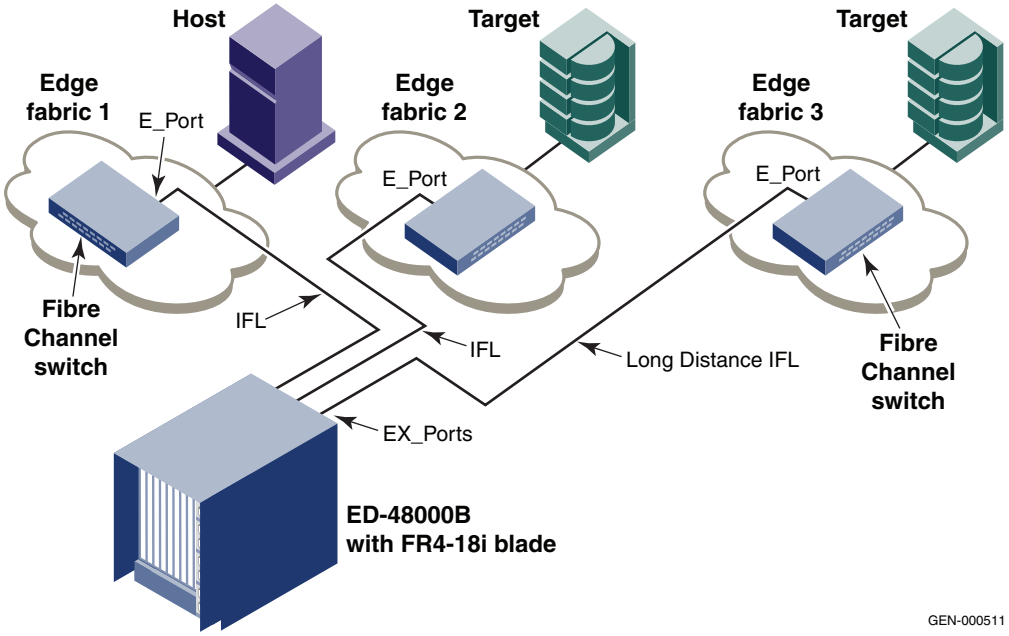


Figure 174 MetaSAN with interfabric links (IFLs)

Edge fabric

An edge fabric is a Fibre Channel fabric with targets and/or initiators connected through Fibre Channel routers via an EX_Port.

Backbone fabric

A backbone fabric is an intermediate network that connects one or more edge fabrics. A backbone fabric consists of at least one SAN router and possibly a number of Connectrix B Fibre Channel switches. A backbone fabric enables hosts and targets in one edge fabric to communicate with devices in the other edge fabrics connected to different routers in the backbone (see Figure 175 on page 313).

Fabric ID (FID)

Every EX_Port uses the FID property to identify the edge fabric. All of the EX_Ports attached to the same edge fabric must have the same FID. The FID for every edge fabric *must* be unique from each backbone fabric's perspective.

When two different backbones are connected to the same edge fabric, the backbone FIDs are different but each edge fabric has the same FID. Configuring the same FID for two separate backbone fabrics is invalid. In this configuration, an RAS log message displays a warning about a fabric ID overlap. However, if two backbone fabrics are not connected to the same edge, they can have the same FID.

MetaSAN

A MetaSAN is a collection of SAN devices, switches, edge fabrics, Logical Storage Area Networks (LSANs), and Brocade Fibre Channel routers that comprise a physically connected, but logically partitioned, storage network.

A simple MetaSAN can be constructed using a SAN router. Additional SAN routers can be used to increase the available bandwidth between fabrics and to provide redundancy.

Note: In this chapter, the term "MetaSAN" is used for the collection of all SANs interconnected with Fibre Channel routers.

Proxy device

A proxy device is a virtual device presented into a fabric by a SAN router and represents a real device on another fabric. When a proxy device is created in a fabric, the real Fibre Channel device is considered to be imported into this fabric. The presence of a proxy device is required for interfabric device communication. The proxy device appears to the fabric as a real Fibre Channel device, has a name server entry, and is assigned a valid port ID. The port ID is only relevant on the fabric in which the proxy device has been created.

Proxy PID

The proxy PID is the port ID of the proxy device.

Phantom domains

The SAN router emulates two types of phantom domains: front domain and translate domain.

Front domain

There is one front domain from a SAN router to an edge fabric, even if there are multiple IFLs from that SAN router to the same edge fabric.

Translate domain

The SAN router creates a translate domain for devices exported from one edge fabric to another. The translate domain has logical ISL connections to the front domains in each EX_Port that connects to an edge fabric. If the translate domain is in a backbone fabric, then it is topologically present within the backbone because there is no front domain for a routed node within a backbone fabric. The translate domain is a proxy domain used to contain devices imported from another fabric. Devices in different edge fabrics can communicate without merging the edge fabrics using this translate and front domain architecture.

Translate domains are sometimes referred to as *xlate* domains. If a SAN router is attached to an edge fabric using an EX_Port, it will create translate domains in the fabric corresponding to the imported edge fabrics with active LSANs defined. If you import devices into the backbone fabric, then a translate domain is created in the backbone device.

If you lose connectivity to the edge fabric due to link failures or the IFL being disabled, translate domains remain visible. This prevents unnecessary fabric disruptions caused by translate domains repeatedly going offline and online when IFL connections are disrupted. To remove the translate domain in the backbone, disable all EX_Ports through which the translate domains were created.

Figure 175 shows another MetaSAN consisting of a host in edge fabric 1 connected to storage in edge fabric 2 through a backbone fabric connecting two Connectrix ED-48000Bs, each containing FR4-18i SAN routing blades.

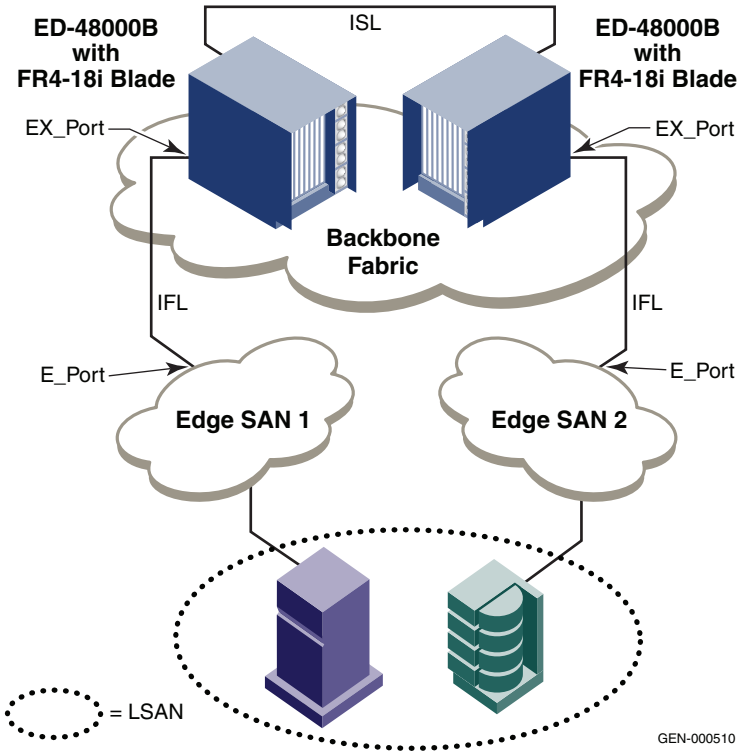


Figure 175 Edge fabrics connected through a backbone fabric

Supported configurations and platforms

In an edge fabric that contains a mix of Administrative Domain (AD) capable switches and switches that are not aware of AD, a device that supports Fibre Channel routing (FCR) should be connected directly to the AD-capable switch.

The supported configurations are:

- SAN router connected to a Brocade FOS fabric
- SAN router connected to a Brocade Secure Fabric OS fabric

Note: EMC does not support Brocade Secure Fabric OS.

- SAN router connected to a Connectrix M Open mode fabric
- SAN router connected to a Connectrix M McDATA Fabric mode (native mode) fabric
- SAN router connected to a FOS and Secure Fabric OS fabrics with EX_Port ISL Trunking enabled
- First generation Fibre Channel routers
  - AP-7420B with Fabric OS v5.1 edge fabrics

Proxy devices

A SAN router achieves interfabric device connectivity by creating proxy devices (hosts and targets) in attached edge fabrics that represent real devices in other fabrics once administrators create the appropriate zones in each edge fabric. For example, a host in Fabric 1 can communicate with a target in Fabric 2 as follows:

- A proxy target in Fabric 1 represents the real target in Fabric 2.
- A proxy host in Fabric 2 represents the real host in Fabric 1.

The host discovers and sends Fibre Channel frames to the proxy target. The SAN router receives these frames and delivers them to the destination fabric for delivery to the target.

The target responds by sending frames to the proxy host. Hosts and targets are exported from the edge fabric to which they are attached and, correspondingly, imported into the edge fabric reached through Fibre Channel routing. Figure 176 on page 315 illustrates this concept.

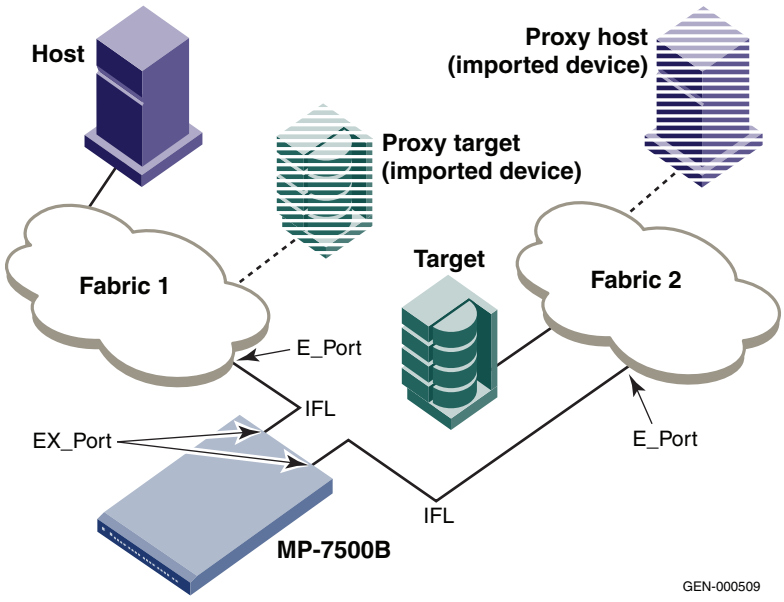


Figure 176 MetaSAN with imported devices

Routing types

The following routing types are discussed in this section:

- "Edge-to-edge," next
- "Backbone-to-edge" on page 315
- "Fibre Channel NAT and phantom domains" on page 316

Edge-to-edge

Fibre Channel routing between edge fabrics occurs when devices in one edge fabric communicate with devices in another edge fabric through one or more Fibre Channel routers.

Backbone-to-edge

This is when one or more Fibre Channel routers form a common fabric, known as a *backbone* fabric (see "Backbone fabric" on page 310). A backbone fabric is created with a standalone SAN router or with SAN routers connected by E_Ports. SAN routers also enable hosts and targets in edge fabrics to communicate with devices in the backbone fabric. This is known as "backbone-to-edge routing." From the edge fabric's perspective, the backbone fabric is just like any other edge fabric. For the edge fabric and backbone fabric devices to

communicate, the shared devices need to be presented to each other's native fabric using the LSAN zoning process. To accomplish this, at least one translate domain is projected into the backbone fabric. This translate domain represents the entire edge fabric. The shared physical device in the edge has a corresponding proxy device attached to the translate domain.

Each edge fabric has only one translate domain for the backbone fabric. The backbone fabric device communicates with the proxy devices whenever it needs to communicate with the shared physical device in the edge. The SAN routing Service receives the frames from the backbone fabric destined to the proxy device, and redirects the frame to the actual physical device. As with an edge fabric, the translate domain switch can never be the principal switch of the backbone fabric. Only translate domains are created in the backbone fabric. (Front domains are not created for backbone-to-edge routing.)

Devices are exported from the backbone fabric to one or more edge fabrics using LSANs.

Fibre Channel NAT and phantom domains

Within an edge fabric or across a backbone fabric, the standard Fibre Channel FSPF protocol determines how frames are routed from the source device to the destination device. The source or destination device can be a proxy device.

Fibre Channel requires that all ports (including EX_Ports) be identified by a unique PID. In a single fabric, FC protocols guarantee that domain IDs are unique, and a PID formed by a domain ID and area ID is unique within a fabric. However, the domain IDs and PIDs in one fabric might be duplicated within another fabric.

SAN routing performs Fibre Channel network address translation (FC-NAT) so the proxy devices in a fabric can have different PIDs than the real devices that they represent. This allows overlapping PIDs in different edge fabrics to communicate and edge fabrics with common domain IDs to have nodes routed between them.

All EX_Ports connected to same edge fabric from one physical FC router present a single front domain and one additional translate (xlate) domain for each edge fabric accessed. All EX_Ports connected to an edge fabric use the same xlate domain ID number for an imported edge fabric. This value persists across switch reboots and fabric reconfiguration. Xlate domains are presented as being connected topologically behind one or more front domains. Each Fibre Channel router presents one front domain per edge fabric. This

allows redundant paths to remote fabrics to present redundant paths to proxy devices to an edge fabric.

Both front and translate domains appear to an edge fabric as logical switches. When an EX_Port is attached to an edge fabric, the edge fabric sees a switch trying to join the fabric via an E_Port. The edge fabric provides a domain ID to the front domain and adds it to its local FSPF routing table. Similarly, the translate domain is added to the edge fabric when it is created. The combination of front domains and translate domains allows routing around path failures, including path failures through the routers. The multiple paths to a translate domain provide additional bandwidth and redundancy.

There are some differences in how the translate domain is presented in the backbone fabric. The backbone translate domains are topologically connected to SAN routers and participate in FC protocol in the backbone. Front domains are not needed in the backbone fabric. As in the case of translate domains for an edge fabric, backbone translate domains provide additional bandwidth and redundancy by being able to present itself being connected to single or multiple SAN routers with each SAN router capable of connecting multiple IFLs to edge fabrics.

DWDM

Dense Wavelength Division Multiplexing (DWDM) is a process in which different channels of data are carried at different wavelengths over one pair of fiber-optic links. This is in contrast with a conventional fiber-optic system in which just one channel is carried over a single wavelength traveling through a single fiber.

CWDM

Coarse Wave Division Multiplexing (CWDM), like DWDM, uses similar processes of multiplexing and de-multiplexing different channels by assigning different wavelengths to each channel. CWDM is intended to consolidate environments containing a low number of channels at a reduced cost.

FastWrite

The FastWrite feature uses the nature of SCSI I/O (specifically the transfer ready phase) to reduce the total command completion time by tricking the initiator into sending data earlier than it normally would.

Figure 177 depicts how a typical "local" SCSI I/O is performed.

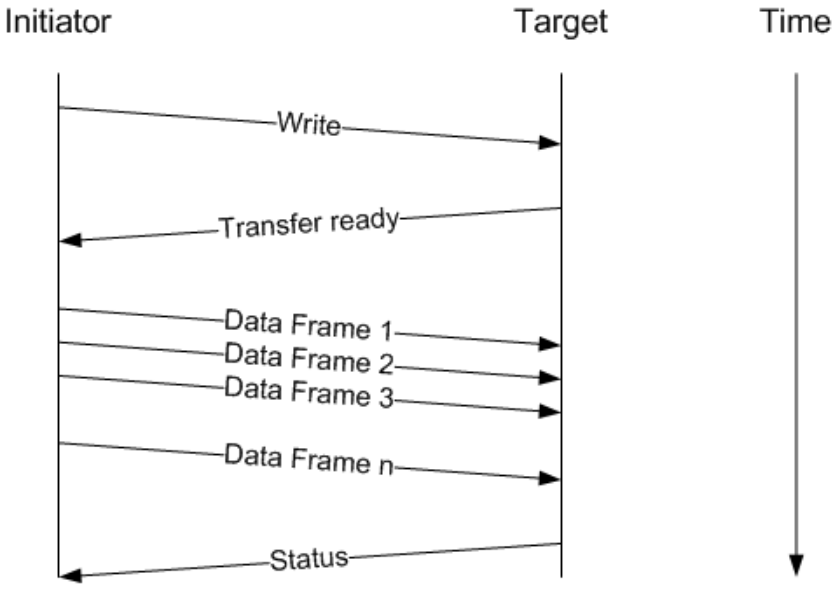


Figure 177 Typical SCSI WRITE

The **WRITE** command is sent and the target responds with *transfer ready*. The amount of data that the target allows the initiator to send is specified inside the transfer ready.

Figure 178 depicts a SCSI I/O being performed over a longer distance. Note that the long delay from the time the command is sent until the time the first data is sent.

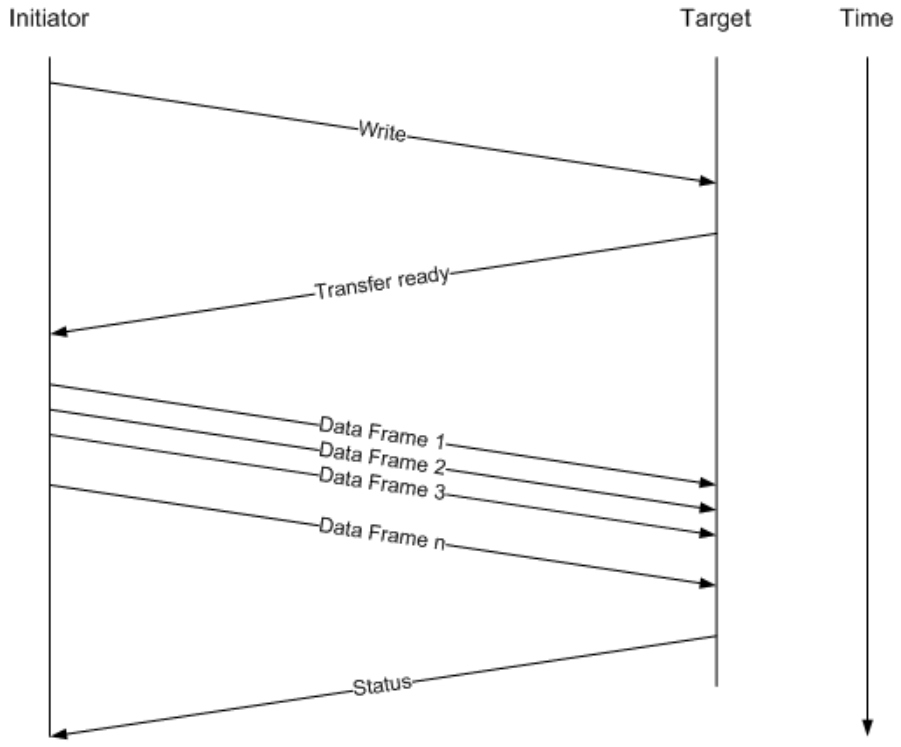


Figure 178 SCSI WRITE over distance without FastWrite

Figure 179 depicts a SCSI I/O being performed over the same distance as in Figure 178, but in this case an appliance that supports FastWrite is inserted into the data path. Note that the amount of time saved in this I/O is approximately 1/2 the roundtrip time.

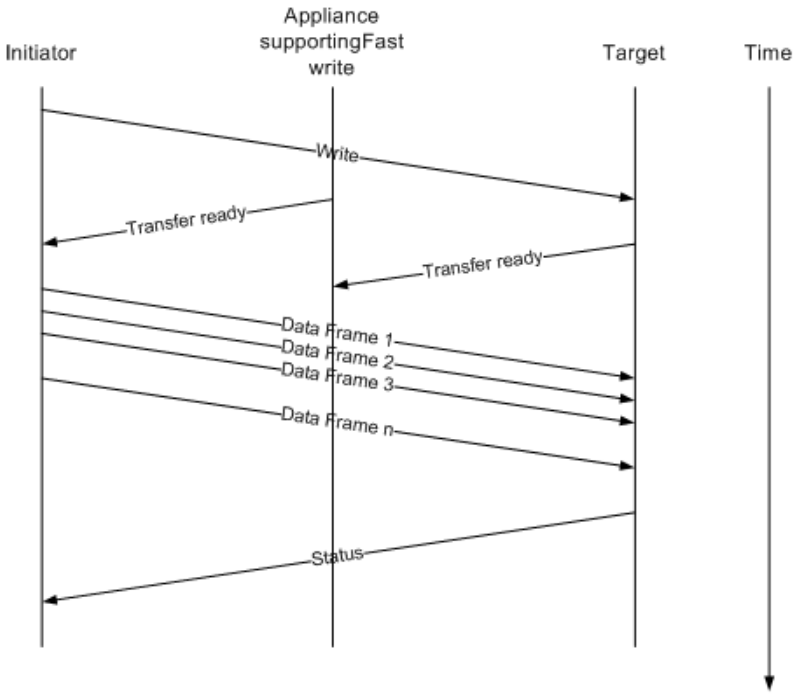


Figure 179 FastWrite over distance with appliance

The amount of time saved can be significant, but before any time savings can be realized, the latency added by the device needs to be overcome. This means that for short distances, enabling FastWrite may *negatively* impact performance. The performance benefit that your environment will realize from using FastWrite depends not only on the distance, but on the applications using the link as well.

Contrary to some information being provided by a switch vendor, a FastWrite feature has not been built into SRDF. However, due to the nature of the I/O being performed, some types of SRDF will see a different result when using an appliance that supports FastWrite.

SRDF over FC and MirrorView

With SRDF over FC, there are always two round trips per write. With MirrorView, degradation due to distance can be more pronounced since each write results in multiple I/Os being created. In both cases, some benefit will be realized using FastWrite at intermediate and long distances:

- 30-60 km with Cisco's write acceleration using their Cisco MDS w/ SSM blade
- 100 km and greater with Brocade M-2640 or M-1620's FastWrite feature

Note: The reason that Cisco provides a benefit starting at 30-60 km and McDATA does not provide a benefit until 100 km relates to differences in internal efficiencies.

Adaptive Copy

All environments using Adaptive Copy will benefit from a FastWrite appliance. Although improvement increases with distance, a good average is 20-30%.

SRDF/A

FastWrite helps to minimize the impact of gaps in data over long distances. Since SRDF/A stream writes back-to-back, there normally are no gaps. As a result, since it never waits for a response before putting more data on the link, SRDF/A *rarely* benefits from FastWrite. EMC does, however, still recommend enabling FastWrite for SRDF, because it has significant benefit during both initial R1-R2 syncs and during a resync after SRDF/A suspension (planned or unplanned) and it does not affect SRDF/A performance.

The exceptions would be small writes or writes to only few volumes, where SRDF/A performance is throttled by RF max I/O rate or the limit of 32 concurrent writes per volume. In these cases, there would be data gaps, and command response time (roundtrip) directly affects throughput.

Vendor-specific features

This section contains information about some of the more important vendor-specific features: partitions, virtual switches, VSANs, IVR (Inter-VSAN Routing), and IVR-NAT (Inter-VSAN Routing Network Address Translation).

Partitions

Partitions were first introduced on the ED-10000M. Each partition in a chassis can be thought of as a separate physical switch with a couple of exceptions:

- They share the same CTPs — This can become a problem if a single port in the same or another partition were to malfunction and inundate the switch with Login or Name Server requests. All partitions would equally share the load and this could result in elongated service times on partitions that would otherwise be isolated from fabric events on another partition.
- They share the same SWMs — This can be a problem in high usage environments using LMQs. The partition a port belongs to is not considered in the fairness algorithm. Therefore, all ports in the same quadrant share the available bandwidth equally, regardless of the partition they are in.
- They share the same power and cooling components — This is a potential availability issue.
- They must all be running the same version of firmware.

There can be up to four partitions in a single chassis with each one consisting of some number of line cards. Each partition must contain at least one line card. No internal routing function exists between two partitions. This means that in order for two ports in different partitions to access each other, an ISL needs to be connected between them.

Virtual switches

Virtual switches were first introduced by Brocade M Series on the ED-10000M with firmware 9.1. A virtual switch is very similar to a VSAN with a few notable exceptions:

- Only four virtual switches are supported per partition with 9.1 and 9.2.
- Port Channels are not supported.

VSANs

Virtual SANs (VSANs) were first introduced by Cisco on its 9509 and 9216 products with Firmware 1.x. A VSAN is a group of hosts or storage ports that communicate with each other using a virtual topology defined on the physical SAN. Using VSANs, you can build a single topology containing switches, links, and one or more VSANs. Each VSAN in this topology has the same behavior and property of a SAN.

A VSAN has the following additional features:

- Multiple VSANs can share the same physical topology.
- The same Fibre Channel IDs (FC IDs) can be assigned to a host in another VSAN, thus increasing VSAN scalability.
- Every instance of a VSAN runs all required protocols such as FSPF, domain manager, and zoning.
- Fabric-related configurations in one VSAN do not affect the associated traffic in another VSAN.
- Events causing traffic disruptions in one VSAN are contained within that VSAN and are not propagated to other VSANs.
- VSANs require that both switches have the VSAN defined on them in order for them to access each other across the ISL. In other words, if Cisco switch A has 10 N_Ports in VSAN 1000, and Cisco switch B has 10 N_Ports in VSAN 1 (default), these N_Ports will not be able to access each other until the N_Ports in VSAN 1 are moved into VSAN 1000, or vice versa, or some flavor of IVR is used.

IVR (Inter-VSAN Routing)

Initiators and targets are allowed to access each other on different VSANs without merging VSANs into a single fabric.
Switch-to-switch control frames (HLO, LSU, BF) do not flow between VSANs, nor can initiators access any resource across VSANs aside from the designated ones.

Virtual SANs (VSANs) improve storage area network (SAN) scalability, availability, and security by allowing multiple Fibre Channel SANs to share a common physical infrastructure of switches and ISLs. These benefits are derived from the separation of Fibre Channel services in each VSAN and isolation of traffic between VSANs. Data traffic isolation between the VSANs also inherently prevents sharing of resources attached to a VSAN, such as a CLARiiON disk library. Using IVR, you can access resources across VSANs without compromising other VSAN benefits.

Data traffic is transported between specific initiators and targets on different VSANs without merging VSANs into a single logical fabric. Fibre Channel control traffic does not flow between VSANs, nor can initiators access any resource across VSANs other than the designated ones. Valuable resources, such as tape libraries, are easily shared across VSANs without compromise.

IVR is not limited to VSANs present on a common switch. Routes that traverse one or more VSANs across multiple switches can be established, if necessary, to establish proper interconnections.

For information on n IVR, refer to the "Cisco Inter VSAN Routing (IVR) in a heterogeneous environment" section in the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

IVR-NAT (Inter-VSAN Routing Network Address Translation)

IVR-NAT allows you to set up IVR in a fabric without needing unique Domain IDs on every switch in the IVR path. When IVR-NAT is enabled, the virtualized end device that appears in the native VSAN uses a virtual Domain ID that is unique to the native VSAN. For more information on IVRR-NAT, refer to "Configure IVR with Network Address Translation (NAT)" section in the "Complex Fibre Channel SAN topologies" chapter in the *Fibre Channel SAN Topologies TechBook*, located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

Port fencing

Port fencing is a policy-based feature that allows you to protect your SAN from repeated operational or security problems experienced by switch ports. Port fencing allows you to set threshold limits on the number of specific port events permitted during a given time period. If the port generates more events during the specified time period, the Connectrix Manager (Port fencing feature) blocks the port, disabling transmit and allows you to receive traffic until you have time to investigate, solve the problem, and manually unblock the port.

Threshold alerts

When you launch the Element Manager from a core switch icon and select **Configure > Threshold Alerts**, the **Threshold Alerts** dialog box displays data on threshold alerts received and transmitted. It also provides the full ability to activate, deactivate, create, edit, and delete threshold alerts.

When the Element Manager is launched from a virtual switch icon and you select **Configure > Threshold Alerts**, the **Threshold Alerts** dialog box displays data on threshold alerts received and transmitted. However, function buttons are replaced with only a **View** button. You cannot activate, deactivate, create, edit, or delete threshold alerts. Selecting the **View** button displays the **Threshold Alerts** dialog box, but you can only view configuration settings for current threshold alerts. You cannot change any of these settings.

Management

For more information on SAN management, refer to the *SAN Management Concepts TechBook* located on the E-Lab Interoperability Navigator, **PDFs and Guides** tab.

Public versus private

Today, the decision to put a switch on a public or private network is more dependent on customer requirements than anything else. Some customers like to have direct access to their switches so they can get SNMP traps directly from the switches, while others like the security that a private network affords.

Historically, EMC recommended that all switches be placed on a private network that is isolated from the rest of the data center LAN. This was recommended for a number of reasons, including limiting the scope of testing that needed to be done for each release, as well as being conservative and protecting the switch from the unknown.

There have been cases where all FC switches on a single public data center LAN segment rebooted due to an ARP storm. Although most of the FC switches have been specifically hardened against network problems of this type, it is still recommended to put environments with multiple FC switches either onto separate network segments or onto a single private network that is shielded by the service processor.

4

IP SAN Concepts

This chapter contains the following information:

IP SAN elements

For this discussion, IP can be described in two elements, each discussed in this section:

- "Internetworks" on page 332
- "IP addressing" on page 335

Internetworks

This section contains the following information:

- "Overview" on page 332
- "Transmission Control Protocol (TCP)" on page 333
- "Internetwork routing" on page 334

Overview

An internetwork consists of individual networks joined by routers, as pictured in Figure 180. Networks A and B are connected to create internetwork A/B.

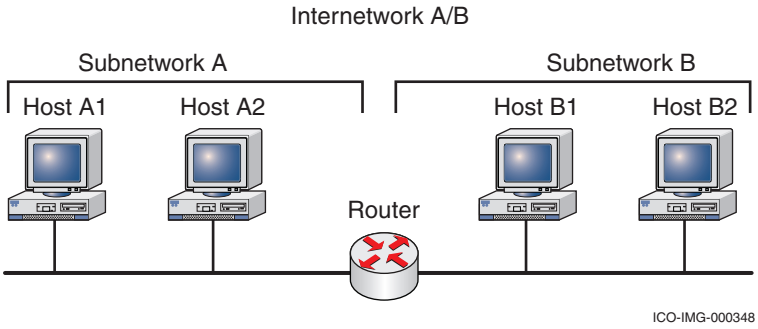


Figure 180 Internetwork example

The Internet is a collection of autonomous networks, each of which contains its own internal network of routers and subnetworks. These autonomous networks are connected to one another using gateway routers and external connections.

Each autonomous network has its own IP network address or range of addresses. Routers are like policemen directing traffic at the network interconnection point. They know the IP network addresses of connected networks and forward packets appropriately.

Every computer on the Internet must have a unique address. IP supports unique addresses by way of a hierarchical addressing scheme. An IP address is a unique number that contains two parts: a network address and host address. The network address is used when forwarding packets across interconnected networks. It defines the destination network and routers along the way know how to forward the packet based on the network address. When the packet arrives at the destination network, the host portion of the IP address identifies the destination host.

IP can be thought of as an addressing scheme for interconnected networks, similar in function to a postal ZIP code which is a sort of overlay-addressing scheme. A ZIP code identifies a specific area, much like an IP address identifies a specific network in a group of interconnected networks. Just as the street address on an envelope defines the destination for a letter, the host portion of the IP address identifies the destination computer.

A design goal of the early Internet was to create a very basic network connectivity model that would provide fast and efficient data exchange among distributed systems without restrictions on the host systems. If applications needed additional communication services, those services would be provided by end systems, and not by the network itself. In fact, the Internet protocols will work on just about any type of underlying network technology, including Ethernet, wireless, and optical.

Transmission Control Protocol (TCP)

No discussion on IP can be complete without at least a reference to the Transmission Control Protocol (TCP), a connection-oriented transport protocol that sends data as an unstructured stream of bytes. By using sequence numbers and acknowledgment messages, TCP provides a transmitting node with delivery information about packets sent to a receiving node. Where data has been lost in transit, TCP can retransmit the data until either a time-out condition is reached or until successful delivery has occurred. TCP can also identify duplicate messages and will discard them as necessary. If the sending computer is transmitting too fast for the receiving computer, TCP can employ flow-control mechanisms to slow data transfer. TCP can also communicate delivery information to the upper-layer protocols and applications it supports

As shown in Figure 181 on page 334, TCP and UDP (User Datagram Protocol) are layered on top of IP and take advantage of IP's datagram delivery services. UDP is a communications protocol that offers a limited amount of services when messages are exchanged

between computers in a network that uses IP. If an application does not need TCP's services, it goes through UDP.

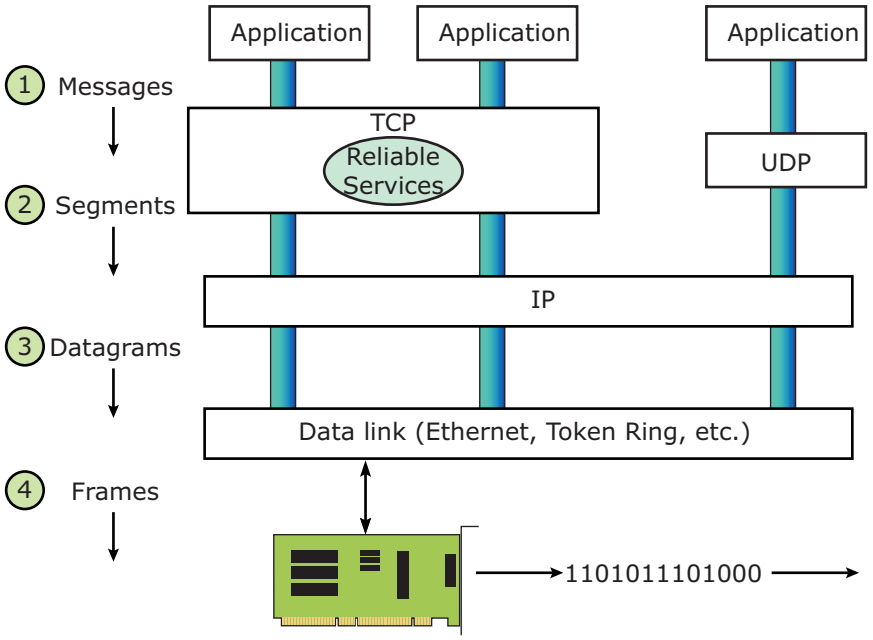


Figure 181 TCP and UDP

Applications make calls to TCP or UDP for network services. Application data or messages are encapsulated in TCP segments or UDP datagrams and then passed down to IP. IP passes the datagrams down to the underlying network, which frames the data into *blocks* for transmission.

Some applications, such as file transfers, require the absolute guarantees that TCP provides in delivering packets. It takes time and additional resources to signal that a packet must be retransmitted, but the delay is worth it to ensure data integrity.

Internetwork routing

Routers (and switches) interconnect different networks through their unique IP address. Routers determine the best way to forward incoming packets. However, a router needs to determine only the best next hop toward a destination, not the complete path to the destination. This is like getting directions while walking in a city. A person might point you in the right direction to an intersection. At the next intersection, another person points you in the right direction.

Eventually, you get to where you want to go—not by knowing the exact path from the start, but by being directed along the way.

At one time, routers were called *gateways* because they provided a path or connection into another network. The term is still used to specify which router on the network serves as a gateway to another network. For example, assume an enterprise LAN is connected by one router to another enterprise LAN, and by another router to the Internet. When configuring an Internet connection, the Internet-connected router is specified as the *primary Internet gateway.*

IP addressing

This section discusses the following:

IP address structure

An IP address is a 32-bit binary number containing two different pieces of information:

- Network identifier — Indicates the network (a group of computers). Networks with different identifiers can be interconnected with routers.
- Host identifier — Indicates a specific computer on the network.

While computers work with IP addresses as 32-bit binary values, humans normally use the dotted-decimal notation. A binary address and its dotted-decimal equivalent are shown below. Note that the 32-bit address is divided into four 8-bit fields called *octets*.

Example: **11000000.10101000.00001010.00000101 = 192.168.10.5**

The abbreviated form makes it easier to discuss the different addressing schemes.

IP address classes

In the early days of the Internet, the 32-bit IP address space was allocated into three classes of addresses:

- Class A — The first 8 bits identify the class and the network, the remaining 24 bits identify hosts. The 24-bit host address space identifies 16,777,214 hosts per each of the 126 networks. Most

class A network schemes were assigned to U.S. government agencies, educational institutions, research organizations and large companies in the early days of the Internet.

- Class B — The first 16 bits identify the class and the network; the remaining 16 bits identify hosts. This scheme defines 16,384 networks and 65,534 hosts per network.
- Class C — The first 24 bits identify the class and the network, the remaining 8 bits identify hosts. This scheme defines 2,097,152 networks and 254 hosts per network.

A class D scheme also exists for multicasting, which is outside the scope of this overview.

The class system would be all but phased out by now except that so many organizations *own* class-based blocks of addresses and many will not voluntarily give them up.

Subnet masks

A subnet mask is an IP address feature that serves as a template to indicate which bits in the IP address define the network and which bits define the host. All devices on the same IP network *must* use the same subnet mask.

A subnet is a logical subsection of an IP network. Subnets are created to separate groups of hosts for many reasons, including security and traffic control. Subnets are usually utilized within enterprise networks to create departmental networks. A service provider with a large block of IP addresses can use subnets to allocate blocks of IP addresses to subscribers. Like networks on the Internet, routers are required at subnet boundaries to transmit packets from one subnet to another.

In terms of addressing, subnetting is equivalent to adding a third level to the Internet addressing hierarchy. The normal levels include all the networks of the Internet, and then the hosts on those networks. Subnetting creates a sublevel of networks within each network. An enterprise or an ISP may configure subnetting to divide a large network into two or more smaller networks that are easier to manage and that match the physical and transmission requirements of the underlying networks.

IP storage networking protocol (iSCSI)

iSCSI (Internet Small Computer System Interface) is an IP-based storage networking standard for linking data storage facilities developed by the Internet Engineering Task Force. By transmitting SCSI commands over IP networks, iSCSI can facilitate block-level transfers over intranets (as opposed to the file sharing found in NAS).

The iSCSI architecture is similar to that of a client/server architectural model. In this case the client is an initiator that issues an I/O request and the server is a target (such as a device in a storage system). The iSCSI initiator issues commands to the target, which fulfills the client's request. The target typically has one or more logical unit numbers (LUNs) that process the initiator's commands.

These commands are contained in a Command Descriptor Block (CDB) that has been issued by the host operating system and translated by the iSCSI driver. In the case of a *read*, the target LUN begins transferring the requested blocks back to the initiator. The iSCSI driver then translates the data into a format the host operating system will recognize.

In a typical EMC environment, for example in mid-range Windows 2000 environments, iSCSI will be used to increase access to hosts that may not have justified the cost of a Fibre Channel host bus adapter. Figure 182illustrates operations in a typical iSCSI implementation in an EMC environment with existing devices.

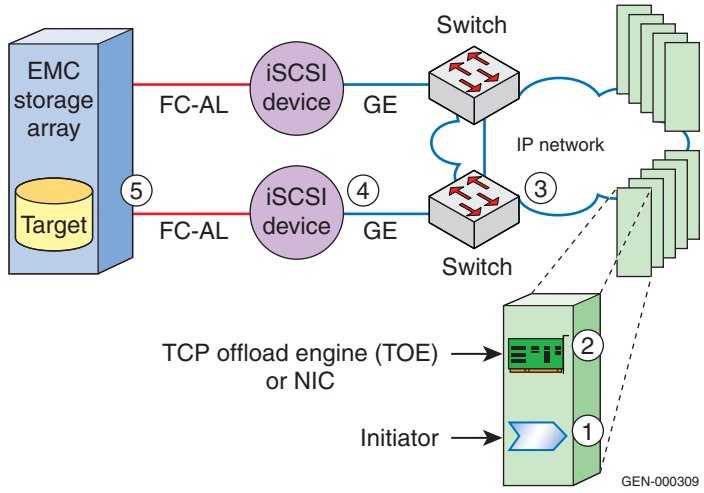


Figure 182 iSCSI implementation example

The process shown in Figure 182 is:

1. Customer application initiates I/O request to the iSCSI *drive*.
2. ISCSI driver encapsulates CDB transaction and transmits I/O over a 10/100 or GE NIC card/TOE engine.
3. Network switch routes packet to appropriate iSCSI router.

4. ISCSI device decodes packet and routes I/O to appropriate LUN.
5. LUN services application request.

You can find a more in-depth review of iSCSI at:

http://www.snia.org/forums/ipsf/programs/about/isci/iSCSI_Technical_whitepaper.PDF

IP over EMC Symmetrix directors

This section describes IP connectivity using channel directors in EMC Symmetrix 8000 and SymmetrixDMX™ series systems:

SRDF over GigE remote director

Symmetrix DMX series systems provide native IP support through Multiprotocol Channel Directors (MPCDs). Symmetrix 8000 series systems provide native IP support through Gigabit Ethernet (GigE) remote directors. The MPCD and GigE remote director provide comparable functionality (with the exception of data compression, which is a feature of the MPCD only).

GigE support for SRDF on Symmetrix systems is based on gigabit Ethernet technology that enables direct Symmetrix-to-IP network attachment. This increases the options for Symmetrix-to-Symmetrix connectivity. It also allows a Symmetrix system to connect to existing Ethernet infrastructure and directly access high-speed data transmission conduits through IP.

Unless otherwise specified, the following information applies to both Symmetrix 8000 series and Symmetrix DMX GigE remote director configurations.

Connectivity

GigE remote director ports are 1000base SX, and can connect to the IP network with 50 micron or 62.5 micron multimode cable. Symmetrix 8000 series systems use dual SC-style connectors; Symmetrix DMX series systems use SFP LC-style connectors.

A Symmetrix system may be connected through GigE remote connections in a one-to-one or a many-to-many (also referred to as fan-in/fan-out) configuration.

When planning an SRDF-over-IP network deployment, it is important to understand the network infrastructure to be used for Symmetrix SRDF connectivity. Network infrastructure can impact TCP connectivity and performance.

TCP connections

SRDF throughput scales per TCP connection; therefore, it is desirable to create more TCP connections through a switched network infrastructure. The number of TCP connections created depends on the network infrastructure between the Symmetrix systems and the mappings of SRDF groups to remote IP targets created in the Symmetrix **IMPL.bin** files.

In the simplest case of a single point-to-point physical connection between a pair of Symmetrix ports (shown in Figure 183), a single TCP connection is created. Scaling to two point-to-point physical connections would allow only two TCP connections.

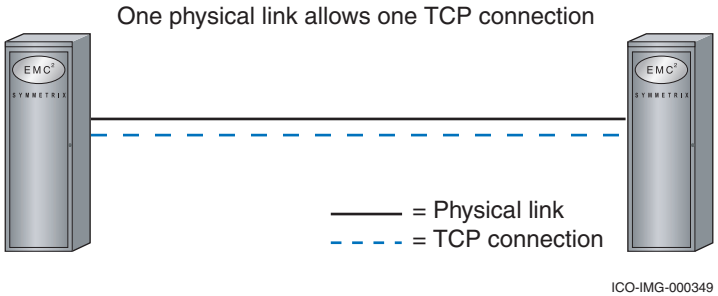


Figure 183 TCP connection example: One TCP connection

With a switched network infrastructure, each Symmetrix port can establish a connection to all remote ports configured as targets. For example, a configuration with two source ports and two remote targets configured (as shown in Figure 184)creates four TCP connections.

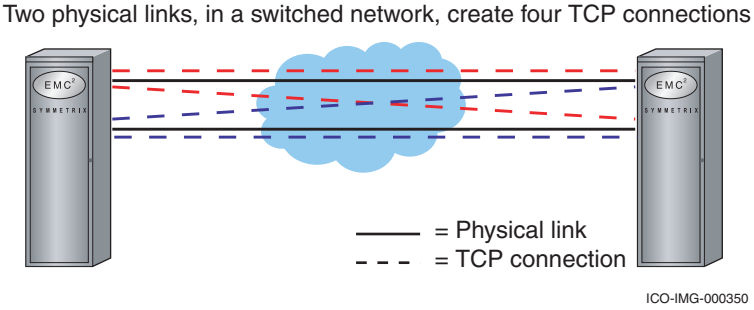


Figure 184 TCP connection example: Four TCP connections

Compression

The Symmetrix DMX MPCD compresses SRDF data prior to transmission on the link. Depending on the compressibility of the data, this can lower network bandwidth requirements. The SRDF data is compressed prior to encapsulation in IP packets.

The MCPD supports compression on a per TCP connection basis. The compression feature should be enabled in the file **IMPL.bin** for source directors and target IP addresses where compression functionality is desired.

Once configured in **IMPL.bin**, the compression feature can be temporarily disabled and re-enabled using Symmetrix **Inlines** commands. To successfully negotiate a TCP connection with compression enabled, both source and target ports must have compression enabled. If compression is not enabled on either end, compression is negotiated to *disabled*.

Symmetrix DMX-to-Symmetrix 8000 connectivity

Symmetrix DMX MPCDs may connect to and establish SRDF connections with Symmetrix 8000 series GigE remote directors. The only feature not available in these configurations is data compression. If data compression is enabled on the MPCD, it will recognize that the peer cannot support compression and the TCP connection will negotiate to compression *disabled*. Symmetrix MPCDs cannot establish a TCP connection with third-party compression devices.

Figure 185 shows both Symmetrix DMX and 8000 series switches connecting to an IP network through switches or routers.

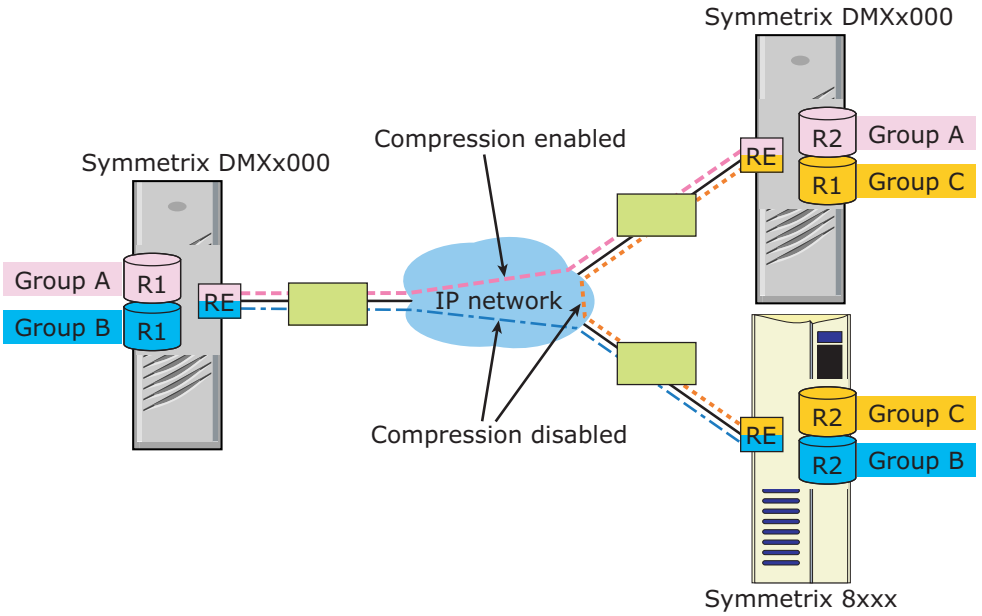


Figure 185 Connectivity: Symmetrix DMX series to Symmetrix 8000 series

iSCSI using Symmetrix multiprotocol channel director

Symmetrix DMX1000, DMX2000, and DMX3000 systems provide high-performance native iSCSI support through the combination of the multiprotocol channel director (MPCD) and the GigE (gigabit Ethernet) back adapter. Symmetrix DMX800 systems also provide native iSCSI support through MPCD to work with DMX800 front-end/back-end adapters (FEBEs).

Supportable configurations

Multiple Ethernet media access types

Multiple media access types are supported through conversion to the optical LC connector over 850 nm multimode shortwave (Figure 186).

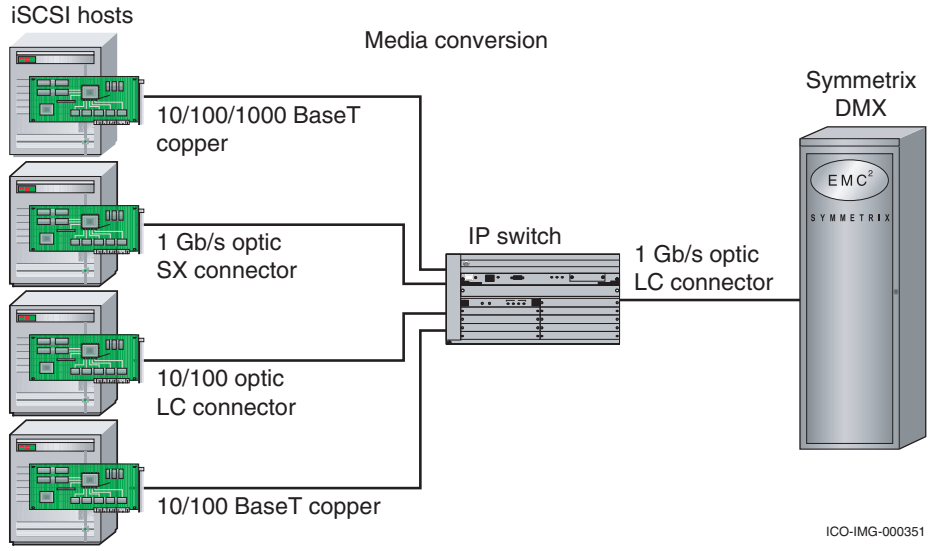


Figure 186 Media conversion

Multiple TCP/IP network topology types

Figure 187 and Figure 188 show the supported TCP/IP network configurations.

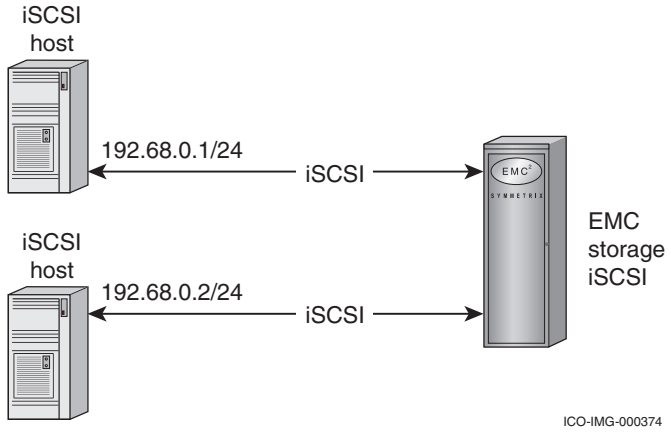


Figure 187 Direct connection of host NIC (from multiple media types) to iSCSI MPCD

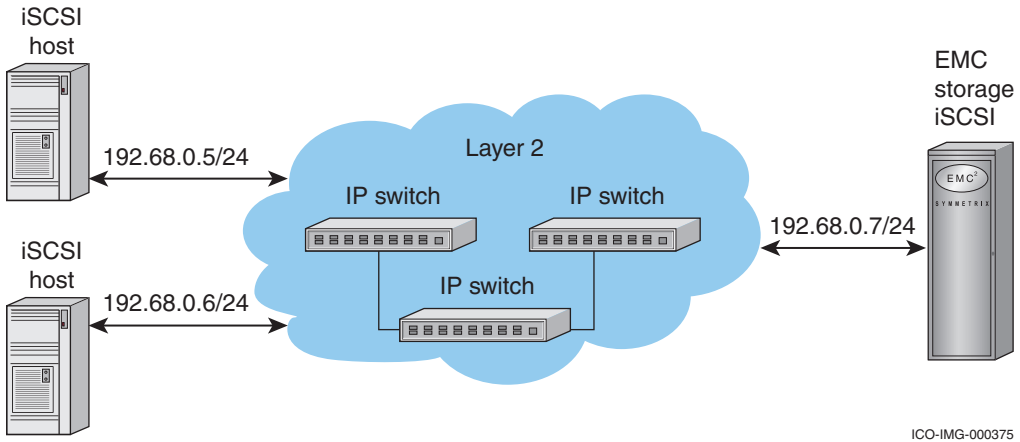


Figure 188 **Switched layer 2 (single subnet) to iSCSI MPCD**

Security implementation

Authentication is implemented through unidirectional CHAP (Challenge Handshake Access Protocol). The credentials and the secret are stored inside the EMC device-masking database.

The EMC *Solutions Enabler Symmetrix Device Masking CLI Product Guide* provides more information.

Data integrity

Symmetrix iSCSI MPCD supports the header digest and data digest using the CRC-32C method. The digest is negotiable by the host that logs in to the Symmetrix system. The *DMX iSCSI Director Technical Notes* provide more information.

Glossary

This glossary contains terms related to EMC products and EMC networked storage concepts.

A

access control A service that allows or prohibits access to a resource. Storage management products implement access control to allow or prohibit specific users. Storage platform products implement access control, often called LUN Masking, to allow or prohibit access to volumes by Initiators (HBAs). *See also* "persistent binding" and "zoning."

active domain ID The domain ID actively being used by a switch. It is assigned to a switch by the principal switch.

active zone set The active zone set is the zone set definition currently in effect and enforced by the fabric or other entity (for example, the name server). Only one zone set at a time can be active.

agent An autonomous agent is a system situated within (and is part of) an environment that senses that environment, and acts on it over time in pursuit of its own agenda. Storage management software centralizes the control and monitoring of highly distributed storage infrastructure. The centralizing part of the software management system can depend on agents that are installed on the distributed parts of the infrastructure. For example, an agent (software component) can be installed on each of the hosts (servers) in an environment to allow the centralizing software to control and monitor the hosts.

alarm An SNMP message notifying an operator of a network problem.

any-to-any port connectivity A characteristic of a Fibre Channel switch that allows any port on the switch to communicate with any other port on the same switch.

application Application software is a defined subclass of computer software that employs the capabilities of a computer directly to a task that users want to perform. This is in contrast to system software that participates with integration of various capabilities of a computer, and typically does not directly apply these capabilities to performing tasks that benefit users. The term application refers to both the application software and its implementation which often refers to the use of an information processing system. (For example, a payroll application, an airline reservation application, or a network application.) Typically an application is installed "on top of" an operating system like Windows or LINUX, and contains a user interface.

application-specific integrated circuit (ASIC) A circuit designed for a specific purpose, such as implementing lower-layer Fibre Channel protocols (FC-1 and FC-0). ASICs contrast with general-purpose devices such as memory chips or microprocessors, which can be used in many different applications.

arbitration The process of selecting one respondent from a collection of several candidates that request service concurrently.

ASIC family Different switch hardware platforms that utilize the same port ASIC can be grouped into collections known as an ASIC family. For example, the Fuji ASIC family which consists of the ED-64M and ED-140M run different microprocessors, but both utilize the same port ASIC to provide Fibre Channel connectivity, and are therefore in the same ASIC family. For inter operability concerns, it is useful to understand to which ASIC family a switch belongs.

ASCII ASCII (American Standard Code for Information Interchange), generally pronounced [aeski], is a character encoding based on the English alphabet. ASCII codes represent text in computers, communications equipment, and other devices that work with text. Most modern character encodings, which support many more characters, have a historical basis in ASCII.

audit log A log containing summaries of actions taken by a Connectrix Management software user that creates an audit trail of changes. Adding, modifying, or deleting user or product administration

values, creates a record in the audit log that includes the date and time.

authentication Verification of the identity of a process or person.

B

backpressure The effect on the environment leading up to the point of restriction. See "congestion."

BB_Credit *See* "buffer-to-buffer credit."

beaconing Repeated transmission of a beacon light and message until an error is corrected or bypassed. Typically used by a piece of equipment when an individual Field Replaceable Unit (FRU) needs replacement. Beaconing helps the field engineer locate the specific defective component. Some equipment management software systems such as Connectrix Manager offer beaconing capability.

BER *See* "bit error rate."

bidirectional In Fibre Channel, the capability to simultaneously communicate at maximum speeds in both directions over a link.

bit error rate Ratio of received bits that contain errors to total of all bits transmitted.

blade server A consolidation of independent servers and switch technology in the same chassis.

blocked port Devices communicating with a blocked port are prevented from logging in to the Fibre Channel switch containing the port or communicating with other devices attached to the switch. A blocked port continuously transmits the off-line sequence (OLS).

bridge A device that provides a translation service between two network segments utilizing different communication protocols. EMC supports and sells bridges that convert iSCSI storage commands from a NIC-attached server to Fibre Channel commands for a storage platform.

broadcast Sends a transmission to all ports in a network. Typically used in IP networks. Not typically used in Fibre Channel networks.

broadcast frames Data packet, also known as a broadcast packet, whose destination address specifies all computers on a network. *See also* "multicast."

buffer Storage area for data in transit. Buffers compensate for differences in link speeds and link congestion between devices.

buffer-to-buffer credit The number of receive buffers allocated by a receiving FC_Port to a transmitting FC_Port. The value is negotiated between Fibre Channel ports during link initialization. Each time a port transmits a frame it decrements this credit value. Each time a port receives an R_Rdy frame it increments this credit value. If the credit value is decremented to zero, the transmitter stops sending any new frames until the receiver has transmitted an R_Rdy frame. Buffer-to-buffer credit is particularly important in SRDF and Mirror View distance extension solutions.

C

Call Home A product feature that allows the Connectrix service processor to automatically dial out to a support center and report system problems. The support center server accepts calls from the Connectrix service processor, logs reported events, and can notify one or more support center representatives. Telephone numbers and other information are configured through the Windows NT dial-up networking application. The Call Home function can be enabled and disabled through the Connectrix Product Manager.

channel With Open Systems, a channel is a point-to-point link that transports data from one point to another on the communication path, typically with high throughput and low latency that is generally required by storage systems. With Mainframe environments, a channel refers to the server-side of the server-storage communication path, analogous to the HBA in Open Systems.

Class 2 Fibre Channel class of service In Class 2 service, the fabric and destination N_Ports provide connectionless service with notification of delivery or nondelivery between the two N_Ports. Historically Class 2 service is not widely used in Fibre Channel system.

Class 3 Fibre Channel class of service Class 3 service provides a connectionless service without notification of delivery between N_Ports. (This is also known as datagram service.) The transmission and routing of Class 3 frames is the same

as for Class 2 frames. Class 3 is the dominant class of communication used in Fibre Channel for moving data between servers and storage and may be referred to as "Ship and pray."

Class F Fibre Channel class of service

Class F service is used for all switch-to-switch communication in a multiswitch fabric environment. It is nearly identical to class 2 from a flow control point of view.

community

A relationship between an SNMP agent and a set of SNMP managers that defines authentication, access control, and proxy characteristics.

community name

A name that represents an SNMP community that the agent software recognizes as a valid source for SNMP requests. An SNMP management program that sends an SNMP request to an agent program must identify the request with a community name that the agent recognizes or the agent discards the message as an authentication failure. The agent counts these failures and reports the count to the manager program upon request, or sends an authentication failure trap message to the manager program.

community profile

Information that specifies which management objects are available to what management domain or SNMP community name.

congestion

Occurs at the point of restriction. See "backpressure."

connectionless

Non dedicated link. Typically used to describe a link between nodes that allows the switch to forward Class 2 or Class 3 frames as resources (ports) allow. *Contrast with* the dedicated bandwidth that is required in a Class 1 Fibre Channel Service point-to-point link.

Connectivity Unit

A hardware component that contains hardware (and possibly software) that provides Fibre Channel connectivity across a fabric. Connectrix switches are example of Connectivity Units. This is a term popularized by the Fibre Alliance MIB, sometimes abbreviated to connunit.

Connectrix management software

The software application that implements the management user interface for all managed Fibre Channel products, typically the Connectrix -M product line. Connectrix Management software is a client/server application with the server running on the Connectrix service processor, and clients running remotely or on the service processor.

Connectrix service processor An optional 1U server shipped with the Connectrix -M product line to run the Connectrix Management server software and EMC remote support application software.

Control Unit In mainframe environments, a Control Unit controls access to storage. It is analogous to a Target in Open Systems environments.

core switch Occupies central locations within the interconnections of a fabric. Generally provides the primary data paths across the fabric and the direct connections to storage devices. Connectrix directors are typically installed as core switches, but may be located anywhere in the fabric.

credit A numeric value that relates to the number of available BB_Credits on a Fibre Channel port. *See*"buffer-to-buffer credit".

D

DASD Direct Access Storage Device.

default Pertaining to an attribute, value, or option that is assumed when none is explicitly specified.

default zone A zone containing all attached devices that are not members of any active zone. Typically the default zone is disabled in a Connectrix M environment which prevents newly installed servers and storage from communicating until they have been provisioned.

Dense Wavelength Division Multiplexing (DWDM) A process that carries different data channels at different wavelengths over one pair of fiber optic links. A conventional fiber-optic system carries only one channel over a single wavelength traveling through a single fiber.

destination ID A field in a Fibre Channel header that specifies the destination address for a frame. The Fibre Channel header also contains a Source ID (SID). The FCID for a port contains both the SID and the DID.

device A piece of equipment, such as a server, switch or storage system.

dialog box A user interface element of a software product typically implemented as a pop-up window containing informational messages and fields for modification. Facilitates a dialog between the user and the application. Dialog box is often used interchangeably with window.

DID An acronym used to refer to either Domain ID or Destination ID. This ambiguity can create confusion. As a result E-Lab recommends this acronym be used to apply to Domain ID. Destination ID can be abbreviated to FCID.

director An enterprise-class Fibre Channel switch, such as the Connectrix ED-140M, MDS 9509, or ED-48000B. Directors deliver high availability, failure ride-through, and repair under power to insure maximum uptime for business critical applications. Major assemblies, such as power supplies, fan modules, switch controller cards, switching elements, and port modules, are all hot-swappable.

The term director may also refer to a board-level module in the Symmetrix that provides the interface between host channels (through an associated adapter module in the Symmetrix) and Symmetrix disk devices. (This description is presented here only to clarify a term used in other EMC documents.)

DNS *See* "domain name service name."

domain ID A byte-wide field in the three byte Fibre Channel address that uniquely identifies a switch in a fabric. The three fields in a FCID are domain, area, and port. A distinct Domain ID is requested from the principal switch. The principal switch allocates one Domain ID to each switch in the fabric. A user may be able to set a Preferred ID which can be requested of the Principal switch, or set an Insistent Domain ID. If two switches insist on the same DID one or both switches will segment from the fabric.

domain name service name Host or node name for a system that is translated to an IP address through a name server. All DNS names have a host name component and, if fully qualified, a domain component, such as *host1.abcd.com*. In this example, *host1* is the host name.

dual-attached host A host that has two (or more) connections to a set of devices.

E

E_D_TOV A time-out period within which each data frame in a Fibre Channel sequence transmits. This avoids time-out errors at the destination Nx_Port. This function facilitates high speed recovery from dropped frames. Typically this value is 2 seconds.

E_Port Expansion Port, a port type in a Fibre Channel switch that attaches to another E_Port on a second Fibre Channel switch forming an Interswitch Link (ISL). This link typically conforms to the FC-SW standards developed by the T11 committee, but might not support heterogeneous inter operability.

edge switch Occupies the periphery of the fabric, generally providing the direct connections to host servers and management workstations. No two edge switches can be connected by interswitch links (ISLs). Connectrix departmental switches are typically installed as edge switches in a multiswitch fabric, but may be located anywhere in the fabric

Embedded Web Server A management interface embedded on the switch's code that offers features similar to (but not as robust as) the Connectrix Manager and Product Manager.

error detect time out value Defines the time the switch waits for an expected response before declaring an error condition. The error detect time out value (E_D_TOV) can be set within a range of two-tenths of a second to one second using the Connectrix switch Product Manager.

error message An indication that an error has been detected. *See also* "information message" *and* "warning message."

Ethernet A baseband LAN that allows multiple station access to the transmission medium at will without prior coordination and which avoids or resolves contention.

event log A record of significant events that have occurred on a Connectrix switch, such as FRU failures, degraded operation, and port problems.

expansionport *See* "E_Port."

explicit fabric login In order to join a fabric, an Nport must login to the fabric (an operation referred to as an FLOGI). Typically this is an explicit operation performed by the Nport communicating with the F_port of the switch, and is called an explicit fabric login. Some legacy Fibre Channel ports do not perform explicit login, and switch vendors perform login for ports creating an implicit login. Typically logins are explicit.

F

FA Fibre Adapter, another name for a Symmetrix Fibre Channel director.

F_Port Fabric Port, a port type on a Fibre Channel switch. An F_Port attaches to an N_Port through a point-to-point full-duplex link connection. A G_Port automatically becomes an F_port or an E-Port depending on the port initialization process.

fabric One or more switching devices that interconnect Fibre Channel N_Ports, and route Fibre Channel frames based on destination IDs in the frame headers. A fabric provides discovery, path provisioning, and state change management services for a Fibre Channel environment.

fabric element Any active switch or director in the fabric.

fabric login Process used by N_Ports to establish their operating parameters including class of service, speed, and buffer-to-buffer credit value.

fabric port A port type (F_Port) on a Fibre Channel switch that attaches to an N_Port through a point-to-point full-duplex link connection. An N_Port is typically a host (HBA) or a storage device like Symmetrix, VNX series, or CLARiiON.

fabric shortest path first (FSPF) A routing algorithm implemented by Fibre Channel switches in a fabric. The algorithm seeks to minimize the number of hops traversed as a Fibre Channel frame travels from its source to its destination.

fabric tree A hierarchical list in Connectrix Manager of all fabrics currently known to the Connectrix service processor. The tree includes all members of the fabrics, listed by WWN or nickname.

failover The process of detecting a failure on an active Connectrix switch FRU and the automatic transition of functions to a backup FRU.

fan-in/fan-out Term used to describe the server:storage ratio, where a graphic representation of a 1:n (fan-in) or n:1 (fan-out) logical topology looks like a hand-held fan, with the wide end toward n. By convention fan-out refers to the number of server ports that share a single storage port. Fan-out consolidates a large number of server ports on a fewer number of storage ports. Fan-in refers to the number of storage ports that a single server port uses. Fan-in enlarges the storage capacity used by a server. A fan-in or fan-out rate is often referred to as just the

n part of the ratio; For example, a 16:1 fan-out is also called a fan-out rate of 16, in this case 16 server ports are sharing a single storage port.

FCP *See* "Fibre Channel Protocol."

FC-SW The Fibre Channel fabric standard. The standard is developed by the T11 organization whose documentation can be found at T11.org. EMC actively participates in T11. T11 is a committee within the InterNational Committee for Information Technology (INCITS).

fiber optics The branch of optical technology concerned with the transmission of radiant power through fibers made of transparent materials such as glass, fused silica, and plastic.

Either a single discrete fiber or a non spatially aligned fiber bundle can be used for each information channel. Such fibers are often called optical fibers to differentiate them from fibers used in non-communication applications.

fibre A general term used to cover all physical media types supported by the Fibre Channel specification, such as optical fiber, twisted pair, and coaxial cable.

Fibre Channel The general name of an integrated set of ANSI standards that define new protocols for flexible information transfer. Logically, Fibre Channel is a high-performance serial data channel.

Fibre Channel Protocol A standard Fibre Channel FC-4 level protocol used to run SCSI over Fibre Channel.

Fibre Channel switch modules The embedded switch modules in the back plane of the blade server. See "blade server" on page 347.

firmware The program code (embedded software) that resides and executes on a connectivity device, such as a Connectrix switch, a Symmetrix Fibre Channel director, or a host bus adapter (HBA).

F_Port Fabric Port, a physical interface within the fabric. An F_Port attaches to an N_Port through a point-to-point full-duplex link connection.

frame A set of fields making up a unit of transmission. Each field is made of bytes. The typical Fibre Channel frame consists of fields: Start-of-frame, header, data-field, CRC, end-of-frame. The maximum frame size is 2148 bytes.

frame header Control information placed before the data-field when encapsulating data for network transmission. The header provides the source and destination IDs of the frame.

FRU Field-replaceable unit, a hardware component that can be replaced as an entire unit. The Connectrix switch Product Manager can display status for the FRUs installed in the unit.

FSPF Fabric Shortest Path First, an algorithm used for routing traffic. This means that, between the source and destination, only the paths that have the least amount of physical hops will be used for frame delivery.

G

gateway address In TCP/IP, a device that connects two systems that use the same or different protocols.

gigabyte (GB) A unit of measure for storage size, loosely one billion (10^9) bytes. One gigabyte actually equals 1,073,741,824 bytes.

G_Port A port type on a Fibre Channel switch capable of acting either as an F_Port or an E_Port, depending on the port type at the other end of the link.

GUI Graphical user interface.

H

HBA *See* "host bus adapter."

hexadecimal Pertaining to a numbering system with base of 16; valid numbers use the digits 0 through 9 and characters A through F (which represent the numbers 10 through 15).

high availability A performance feature characterized by hardware component redundancy and hot-swappability (enabling non-disruptive maintenance). High-availability systems maximize system uptime while providing superior reliability, availability, and serviceability.

hop A hop refers to the number of InterSwitch Links (ISLs) a Fibre Channel frame must traverse to go from its source to its destination.

Good design practice encourages three hops or less to minimize congestion and performance management complexities.

host bus adapter A bus card in a host system that allows the host system to connect to the storage system. Typically the HBA communicates with the host over a PCI or PCI Express bus and has a single Fibre Channel link to the fabric. The HBA contains an embedded microprocessor with on board firmware, one or more ASICs, and a Small Form Factor Pluggable module (SFP) to connect to the Fibre Channel link.

I

I/O *See* "input/output."

in-band management Transmission of monitoring and control functions over the Fibre Channel interface. You can also perform these functions out-of-band typically by use of the ethernet to manage Fibre Channel devices.

information message A message telling a user that a function is performing normally or has completed normally. User acknowledgement might or might not be required, depending on the message. *See also* "error message" *and* "warning message."

input/output (1) Pertaining to a device whose parts can perform an input process and an output process at the same time. (2) Pertaining to a functional unit or channel involved in an input process, output process, or both (concurrently or not), and to the data involved in such a process. (3) Pertaining to input, output, or both.

interface (1) A shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics as appropriate. The concept includes the specification of the connection of two devices having different functions. (2) Hardware, software, or both, that links systems, programs, or devices.

Internet Protocol *See* "IP."

interoperability The ability to communicate, execute programs, or transfer data between various functional units over a network. Also refers to a Fibre Channel fabric that contains switches from more than one vendor.

interswitch link (ISL) Interswitch link, a physical E_Port connection between any two switches in a Fibre Channel fabric. An ISL forms a hop in a fabric.

IP Internet Protocol, the TCP/IP standard protocol that defines the datagram as the unit of information passed across an internet and provides the basis for connectionless, best-effort packet delivery service. IP includes the ICMP control and error message protocol as an integral part.

IP address A unique string of numbers that identifies a device on a network. The address consists of four groups (quadrants) of numbers delimited by periods. (This is called *dotted-decimal* notation.) All resources on the network must have an IP address. A valid IP address is in the form *nnn.nnn.nnn.nnn*, where each *nnn* is a decimal in the range 0 to 255.

ISL Interswitch link, a physical E_Port connection between any two switches in a Fibre Channel fabric.

K

kilobyte (K) A unit of measure for storage size, loosely one thousand bytes. One kilobyte actually equals 1,024 bytes.

L

laser A device that produces optical radiation using a population inversion to provide light amplification by stimulated emission of radiation and (generally) an optical resonant cavity to provide positive feedback. Laser radiation can be highly coherent temporally, spatially, or both.

LED Light-emitting diode.

link The physical connection between two devices on a switched fabric.

link incident A problem detected on a fiber-optic link; for example, loss of light, or invalid sequences.

load balancing The ability to distribute traffic over all network ports that are the same distance from the destination address by assigning different paths to different messages. Increases effective network bandwidth. EMC PowerPath software provides load-balancing services for server IO.

logical volume A named unit of storage consisting of a logically contiguous set of disk sectors.

Logical Unit Number (LUN) A number, assigned to a storage volume, that (in combination with the storage device node's World Wide Port Name (WWPN)) represents a unique identifier for a logical volume on a storage area network.

M

MAC address Media Access Control address, the hardware address of a device connected to a shared network.

managed product A hardware product that can be managed using the Connectrix Product Manager. For example, a Connectrix switch is a managed product.

management session Exists when a user logs in to the Connectrix Management software and successfully connects to the product server. The user must specify the network address of the product server at login time.

media The disk surface on which data is stored.

media access control *See* "MAC address."

megabyte (MB) A unit of measure for storage size, loosely one million (10^6) bytes. One megabyte actually equals 1,048,576 bytes.

MIB Management Information Base, a related set of objects (variables) containing information about a managed device and accessed through SNMP from a network management station.

multicast Multicast is used when multiple copies of data are to be sent to designated, multiple, destinations.

multiswitch fabric Fibre Channel fabric created by linking more than one switch or director together to allow communication. *See also* "ISL."

multiswitch linking Port-to-port connections between two switches.

N

name server (dNS) A service known as the distributed Name Server provided by a Fibre Channel fabric that provides device discovery, path provisioning, and

state change notification services to the N_Ports in the fabric. The service is implemented in a distributed fashion, for example, each switch in a fabric participates in providing the service. The service is addressed by the N_Ports through a Well Known Address.

network address A name or address that identifies a managed product, such as a Connectrix switch, or a Connectrix service processor on a TCP/IP network. The network address can be either an IP address in dotted decimal notation, or a Domain Name Service (DNS) name as administered on a customer network. All DNS names have a host name component and (if fully qualified) a domain component, such as *host1.emc.com*. In this example, *host1* is the host name and *EMC.com* is the domain component.

nickname A user-defined name representing a specific WWxN, typically used in a Connectrix -M management environment. The analog in the Connectrix -B and MDS environments is alias.

node The point at which one or more functional units connect to the network.

N_Port Node Port, a Fibre Channel port implemented by an end device (node) that can attach to an F_Port or directly to another N_Port through a point-to-point link connection. HBAs and storage systems implement N_Ports that connect to the fabric.

NVRAM Nonvolatile random access memory.

O

offline sequence (OLS) The OLS Primitive Sequence is transmitted to indicate that the FC_Port transmitting the Sequence is:

a. initiating the Link Initialization Protocol

b. receiving and recognizing NOS

c. or entering the offline state

OLS *See* "offline sequence (OLS)".

operating mode Regulates what other types of switches can share a multiswitch fabric with the switch under consideration.

operating system Software that controls the execution of programs and that may provide such services as resource allocation, scheduling, input/output control, and data management. Although operating systems are predominantly software, partial hardware implementations are possible.

optical cable A fiber, multiple fibers, or a fiber bundle in a structure built to meet optical, mechanical, and environmental specifications.

OS *See* "operating system."

out-of-band management Transmission of monitoring/control functions outside of the Fibre Channel interface, typically over ethernet.

oversubscription The ratio of bandwidth required to bandwidth available. When all ports, associated pair-wise, in any random fashion, cannot sustain full duplex at full line-rate, the switch is oversubscribed.

P

parameter A characteristic element with a variable value that is given a constant value for a specified application. Also, a user-specified value for an item in a menu; a value that the system provides when a menu is interpreted; data passed between programs or procedures.

password (1) A value used in authentication or a value used to establish membership in a group having specific privileges. (2) A unique string of characters known to the computer system and to a user who must specify it to gain full or limited access to a system and to the information stored within it.

path In a network, any route between any two nodes.

persistent binding Use of server-level access control configuration information to persistently bind a server device name to a specific Fibre Channel storage volume or logical unit number, through a specific HBA and storage port WWN. The address of a persistently bound device does not shift if a storage target fails to recover during a power cycle. This function is the responsibility of the HBA device driver.

port (1) An access point for data entry or exit. (2) A receptacle on a device to which a cable for another device is attached.

port card Field replaceable hardware component that provides the connection for fiber cables and performs specific device-dependent logic functions.

port name A symbolic name that the user defines for a particular port through the Product Manager.

preferred domain ID An ID configured by the fabric administrator. During the fabric build process a switch requests permission from the principal switch to use its preferred domain ID. The principal switch can deny this request by providing an alternate domain ID only if there is a conflict for the requested Domain ID. Typically a principal switch grants the non-principal switch its requested Preferred Domain ID.

principal downstream ISL The ISL to which each switch will forward frames originating from the principal switch.

principal ISL The principal ISL is the ISL that frames destined to, or coming from, the principal switch in the fabric will use. An example is an RDI frame.

principal switch In a multiswitch fabric, the switch that allocates domain IDs to itself and to all other switches in the fabric. There is always one principal switch in a fabric. If a switch is not connected to any other switches, it acts as its own principal switch.

principal upstream ISL The ISL to which each switch will forward frames destined for the principal switch. The principal switch does not have any upstream ISLs.

product (1) Connectivity Product, a generic name for a switch, director, or any other Fibre Channel product. (2) Managed Product, a generic hardware product that can be managed by the Product Manager (a Connectrix switch is a managed product). Note distinction from the definition for "device."

Product Manager A software component of Connectrix Manager software such as a Connectrix switch product manager, that implements the management user interface for a specific product. When a product instance is opened from the Connectrix Manager software products view, the corresponding product manager is invoked. The product manager is also known as an Element Manager.

product name A user configurable identifier assigned to a Managed Product. Typically, this name is stored on the product itself. For a Connectrix switch, the Product Name can also be accessed by an SNMP Manager as the System Name. The Product Name should align with the host name component of a Network Address.

products view The top-level display in the Connectrix Management software user interface that displays icons of Managed Products.

protocol (1) A set of semantic and syntactic rules that determines the behavior of functional units in achieving communication. (2) A specification for the format and relative timing of information exchanged between communicating parties.

R

R_A_TOV *See* "resource allocation time out value."

remote access link The ability to communicate with a data processing facility through a remote data link.

remote notification The system can be programmed to notify remote sites of certain classes of events.

remote user workstation A workstation, such as a PC, using Connectrix Management software and Product Manager software that can access the Connectrix service processor over a LAN connection. A user at a remote workstation can perform all of the management and monitoring tasks available to a local user on the Connectrix service processor.

resource allocation time out value A value used to time-out operations that depend on a maximum time that an exchange can be delayed in a fabric and still be delivered. The resource allocation time-out value of (R_A_TOV) can be set within a range of two-tenths of a second to 120 seconds using the Connectrix switch product manager. The typical value is 10 seconds.

S

SAN *See* "storage area network (SAN)."

segmentation A non-connection between two switches. Numerous reasons exist for an operational ISL to segment, including interop mode incompatibility, zoning conflicts, and domain overlaps.

segmented E_Port E_Port that has ceased to function as an E_Port within a multiswitch fabric due to an incompatibility between the fabrics that it joins.

service processor *See* "Connectrix service processor."

session *See* "management session."

single attached host A host that only has a single connection to a set of devices.

small form factor pluggable (SFP) An optical module implementing a shortwave or long wave optical transceiver.

SMTP Simple Mail Transfer Protocol, a TCP/IP protocol that allows users to create, send, and receive text messages. SMTP protocols specify how messages are passed across a link from one system to another. They do not specify how the mail application accepts, presents or stores the mail.

SNMP Simple Network Management Protocol, a TCP/IP protocol that generally uses the User Datagram Protocol (UDP) to exchange messages between a management information base (MIB) and a management client residing on a network.

storage area network (SAN) A network linking servers or workstations to disk arrays, tape backup systems, and other devices, typically over Fibre Channel and consisting of multiple fabrics.

subnet mask Used by a computer to determine whether another computer with which it needs to communicate is located on a local or remote network. The network mask depends upon the class of networks to which the computer is connecting. The mask indicates which digits to look at in a longer network address and allows the router to avoid handling the entire address. Subnet masking allows routers to move the packets more quickly. Typically, a subnet may represent all the machines at one geographic location, in one building, or on the same local area network.

switch priority Value configured into each switch in a fabric that determines its relative likelihood of becoming the fabric's principal switch.

T

TCP/IP Transmission Control Protocol/Internet Protocol. TCP/IP refers to the protocols that are used on the Internet and most computer networks. TCP refers to the Transport layer that provides flow control and connection services. IP refers to the Internet Protocol level where addressing and routing are implemented.

toggle To change the state of a feature/function that has only two states. For example, if a feature/function is *enabled*, toggling changes the state to *disabled*.

topology Logical and/or physical arrangement of switches on a network.

trap An asynchronous (unsolicited) notification of an event originating on an SNMP-managed device and directed to a centralized SNMP Network Management Station.

U

unblocked port Devices communicating with an unblocked port can log in to a Connectrix switch or a similar product and communicate with devices attached to any other unblocked port if the devices are in the same zone.

Unicast Unicast routing provides one or more optimal path(s) between any of two switches that make up the fabric. (This is used to send a single copy of the data to designated destinations.)

upper layer protocol (ULP) The protocol user of FC-4 including IPI, SCSI, IP, and SBCCS. In a device driver ULP typically refers to the operations that are managed by the class level of the driver, not the port level.

URL Uniform Resource Locater, the addressing system used by the World Wide Web. It describes the location of a file or server anywhere on the Internet.

V

virtual switch A Fibre Channel switch function that allows users to subdivide a physical switch into multiple virtual switches. Each virtual switch consists of a subset of ports on the physical switch, and has all the properties of a Fibre Channel switch. Multiple virtual switches can be connected through ISL to form a virtual fabric or VSAN.

virtual storage area network (VSAN) An allocation of switch ports that can span multiple physical switches, and forms a virtual fabric. A single physical switch can sometimes host more than one VSAN.

volume A general term referring to an addressable logically contiguous storage space providing block IO services.

VSAN Virtual Storage Area Network.

W

warning message An indication that a possible error has been detected. *See also* "error message" *and* "information message."

World Wide Name (WWN) A unique identifier, even on global networks. The WWN is a 64-bit number (XX:XX:XX:XX:XX:XX:XX:XX). The WWN contains an OUI which uniquely determines the equipment manufacturer. OUIs are administered by the Institute of Electronic and Electrical Engineers (IEEE). The Fibre Channel environment uses two types of WWNs; a World Wide Node Name (WWNN) and a World Wide Port Name (WWPN). Typically the WWPN is used for zoning (path provisioning function).

Z

zone An information object implemented by the distributed Nameserver (dNS) of a Fibre Channel switch. A zone contains a set of members which are permitted to discover and communicate with one another. The members can be identified by a WWPN or port ID. EMC recommends the use of WWPNs in zone management.

zone set An information object implemented by the distributed Nameserver (dNS) of a Fibre Channel switch. A Zone Set contains a set of Zones. A Zone Set is activated against a fabric, and only one Zone Set can be active in a fabric.

zonie A storage administrator who spends a large percentage of his workday zoning a Fibre Channel network and provisioning storage.

zoning Zoning allows an administrator to group several devices by function or by location. All devices connected to a connectivity product, such as a Connectrix switch, may be configured into one or more zones.

Index

G

H

I

L

M

U

Z