The Complete Reference

Part IV

Storage Area Networks

The Complete Reference

Chapter 13

Architecture Overview

In 1993, the largest data warehouse application supported only 50GB of aggregate data. Although this appears trivial by today's standards, it drove the client/server and mainframe computing infrastructures to support larger and larger data capacities. Both mainframe and open server vendors responded by increasing the number of processors, channels, RAM memory, and bus capabilities. However, these technologies had reached a level of maturity in their development that limited dramatic improvements in many of these areas. Coupled with the momentum of online transactional workloads driven by Internet access, web applications, and more robust Windows-based client/server applications, the network began to play an increasing role in supporting the larger data-centric platforms. However, the distribution of work through the network only found the LAN to be challenged by its own data-centric traffic congestion.

In a different response to this challenge, both new and established vendors moved toward hybrid types of computing platforms, while methods appeared to handle the growing data-centric problem (see Chapter 8). These were parallel processing and high-end SMP computing platforms using high-performance relational database software that supported increased degrees of parallel processing. Vendors of systems and databases were thus caught in a dilemma, given that applications requiring data storage above 100GB capacities needed very costly high-end solutions.

Driven by the sheer value of access to data, end-user appetites for data-centric solutions continued to grow unabated. However, these requirements proved increasingly difficult for IT organizations as they struggled to supply, manage, and integrate hybrid solutions into their data centers. Obviously, one of the main areas experiencing dynamic growth within this maelstrom of datacentric activity was storage and storage-related products. This was an interesting dilemma for the storage vendors, given that the key to any enhancement of the data-centric application rollout revolved around a high-end I/O system. This drove storage vendors to the same traditional solutions as their mainframe and open-server brethren: enhance the existing infrastructure of disk density, bus performance (SCSI, PCI), and array scalability.

As the conservative approach from mainframe and open server vendors drove innovators offering alternative solutions to start their own companies and initiatives, a similar evolution began within storage companies. Existing storage vendors stuck to their conservative strategy to enhance existing technologies, spawning new initiatives in the storage industry.

Several initiatives studied applying a network concept to storage infrastructures that allowed processing nodes (for example, servers) to access the network for data. This evolved into creating a storage network from existing architectures and technologies, and interfacing this with existing I/O technologies of both server and storage products. Using the channel-oriented protocol of Fibre Channel, a network model of packet-based switching (FC uses frames in place of packets, however), and a specialized operating environment using the micro-kernel concept, the architecture of Storage Area Networks came into being.

This was such a major shift from the traditional architecture of directly connecting storage devices to a server or mainframe that an entire I/O architecture was turned upside down, prompting a major paradigm shift. This shift is not to be overlooked, or taken lightly, because with any major change in how things function, it must first be understood, analyzed, and evaluated before all its values can be seen. With Storage

Area Networks, we are in that mode. Not that the technology is not useful today. It can be very useful. However, the full value of Storage Area Networks as the next generation of I/O infrastructures still continues to evolve.

Creating a network for storage affects not only how we view storage systems and related products, but also how we can effectively use data within data-centric applications still in demand today. Here we are, ten years from the high-end data warehouse of 50GB, with today's applications supporting 500GB on average. This is only a ten-fold improvement. What can we achieve if we move our data-centric applications into an I/O network developed specifically for storage (perhaps a hundred-fold improvement)? Hopefully, ten years from now, this book will reflect the average database supporting 5 terabytes and being shared among all the servers in the data center.

Creating a Network for Storage

Network attached storage is very different from a Storage Area Network on many levels. First, SANs denote an entire infrastructure supporting storage systems. Secondly, (and maybe this should actually be first) SANs function on their own network developed specifically for shared I/O processing, enhanced storage devices, and scalability within a data center infrastructure. Thirdly, SANs operate on a protocol different than NAS (NAS integrates into traditional TCP/IP networks and associated network topologies) by using Fibre Channel. Lastly, SANs offer complete flexibility within their infrastructure by maintaining the intrinsic I/O communications protocols inherent with directly attached storage.

Whereas NAS remains a retrofit to existing computer networks, SANs offer an entire I/O infrastructure that breaks the storage boundaries of traditional storage I/O models and moves the support for data-centric applications into a new generation of computer processing models.

Several things happen when you create a storage network. The storage becomes accessible through a network model—for example, nodes logging in to the network can communicate with other nodes within the network. Nodes operating within the network offer a diverse amount of function depending on their device types—they can be storage devices, servers, routers, bridges, and even other networks. You can transfer data faster, with more throughput, and with increased flexibility. Managing the resources in a storage network can be performed from a centralized perspective across sharable domains.

This results in an inherent value to the Storage Area Network. Servers can now share the resources of storage systems such as disk arrays and devices. Conversely, storage arrays can be shared among the servers consolidating the number of devices required. This means increased access to centralized data by large numbers of applications, with support for larger storage capacities through increased ways of providing I/O operations.

Storage Area Networks are made up of four major parts. As we discussed with NAS, these parts cover the major areas of I/O operations, storage systems, and supported workloads. In SAN technology, however, there are more seemingly disparate parts that must be integrated to form an entire solution. With NAS, most of the products available are offered as bundled solutions. Not so with SANs. When considering SAN infrastructure, we must ponder more carefully the separate components that make up the infrastructure,

because each operates independently and interdependently with the Storage Area Network they participate in.

Given that consideration, we can discuss and consider the major components of a Storage Area Network and resulting architecture. SAN components include the following:

- **Network Part** SANs provide a separated network over and above what the client/server or mainframe networks utilize in connecting clients, terminals, and other devices, including NAS. This network, as mentioned, is based on the Fibre Channel protocol and standard.
- **Hardware Part** SANs depend on specific hardware devices just like other networks. Given that SANs provide a packet-switched network topology, they adhere to a standard layer of processing that the hardware devices operate within. Again, to avoid confusion between data communications packet topologies, the FC protocol relies on frames and an architecture more compatible with I/O channel operations.
- **Software Part** SANs operate within a separate set of software called a fabric that makes up the FC network. Additional software is required at the server connection where the average Windows or UNIX OS drivers must communicate within a SAN environment. The same thing is true for storage devices that must communicate with the SAN network to provide data to the servers.
- **Connectivity Part** Much of a SAN's value derives from its connectivity, which is made up of several hardware and software functions. Given the complete shift in I/O operations from bus level communications to network communications, many components must change or be modified to operate within this environment. Connectivity options within the SAN determine performance and workload applicability.

Figure 13-1 offers a glimpse into the SAN infrastructure.

The Network Part

SAN is a network and, as such, its main task is to provide communications among devices attached to it. It does so through the FC standard protocol, denoted by the FC standard and relative T-11 standards committee that maintains the protocol. The FC protocol offers a layered approach to communications similar to TCP/IP, but through a smaller set of processing layers. The FC layers are illustrated in Figure 13-2, showing functions FC-0 through FC-4. (More detail on FC layers can be found in Chapter 16.) However, what's important is that each layer is leveraged through different hardware components within the SAN.

What facilitates the network is the switch component that provides a set of circuits making up the communications paths for the devices attached. The communications paths are accommodated through the addressing associated with the devices' communications layers. Therefore, the FC switch device can provide an any-to-any connectivity matrix that allows communications from one device to another.

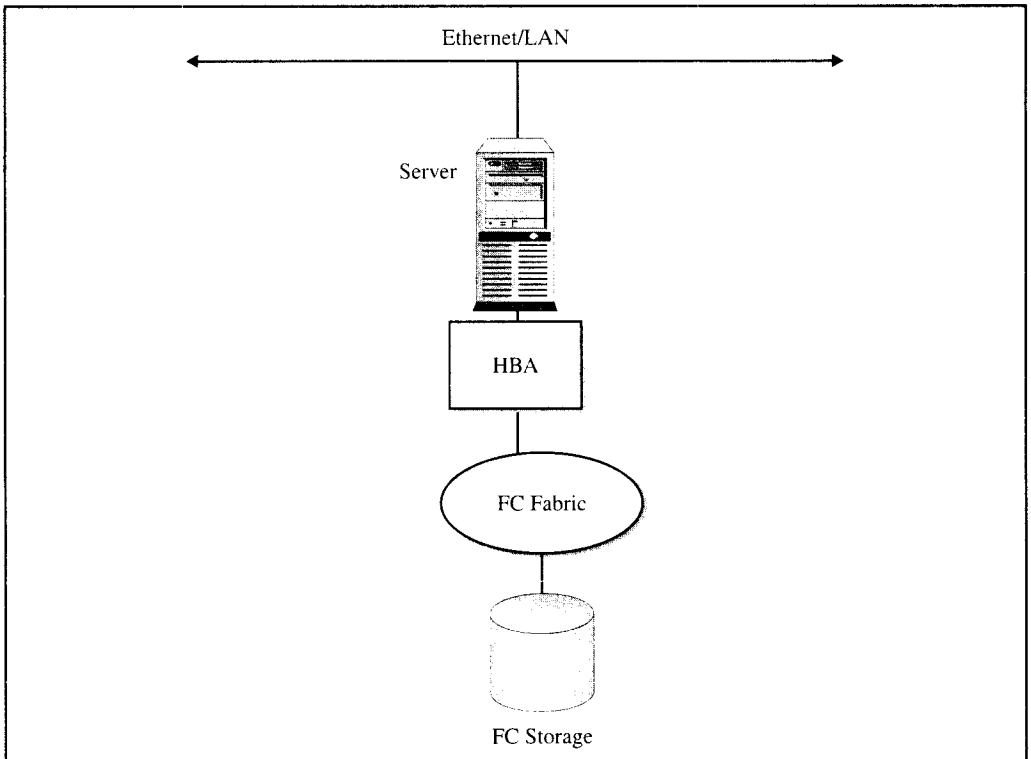


Figure 13-1. *SAN components overview*

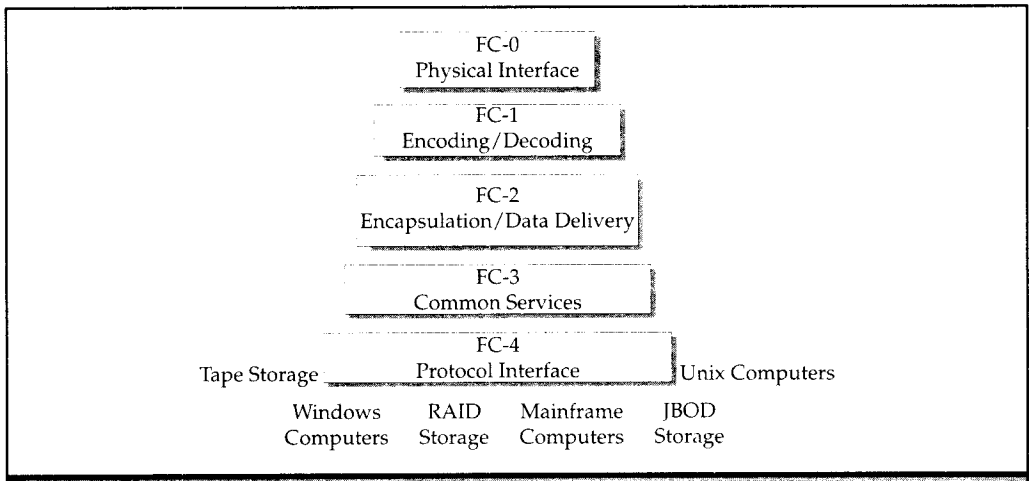


Figure 13-2. *FC layers and functions*

For example, a server may require a read from a particular disk. The server's I/O request provides the address for the requested device, the network fabric develops the connection, and that passes the read I/O to the storage device. Conversely, the storage responds with the same type of process, whereby the storage device requesting a connection with the server address returns the block of data from the read operation, the switch fabric makes the connection, and the I/O operation is completed.

It is important to note that one of the particular values of SANs, and specifically Fibre Channel, is its capability to provide a way for existing protocols to be encapsulated within the communications. This is especially valuable, as the SCSI commands for disk operations do not have to change. Consequently, disk and driver operations can operate within the FC protocol unchanged.

However, it is also important to note that in developing the SAN network, additional components are required beyond the FC switch for both the server and the storage devices. These are depicted in Figure 13-3, as we see the effects of our read I/O operation within the switch. These components—HBAs, FC switches, hubs, and routers—will be discussed in more detail in the section titled "The Hardware Part" in this chapter, as well as in Chapter 14.

Inside the FC network, the fabric software operates on a frame-based design. Similar to concepts used in TCP/IP packets, communications within the network are performed by transmitting FC frames throughout the fabric. The frames are composed of header, user data, and trailer information. FC layers determine the operations responsible for the frame development, frame addressing, and transmission (see Figure 13-3).

We would be remiss if we did not mention other ways the FC network can be configured. This is important because many of these configurations are still operating and some offer specific solutions to particular situations; they're also cost-effective. However, most can be attributed to implementations that occurred prior to the development of the FC Switched fabric solutions.

These solutions include the following:

- **Point-to-Point** This uses the FC to connect one device to another. Employed in initial FC implementations of FC disk arrays, it leveraged the increased bandwidth, but remained a direct attached solution. Also, tape devices could be used with an FC point-to-point configuration, increasing the number of drives supported from a single server.
- **Hub** This uses FC to connect in a loop fashion. Basically, the initial network fabric supported an arbitrated loop arrangement whereby, like SCSI bus configurations, the devices were configured in loop topologies which not only shared bandwidth but had to arbitrate for network communications, like a SCSI bus. It leveraged the speed of FC and the capability to place additional disk arrays within the network allowing connection of additional servers. FC-AL, however, provided latency issues that never fully facilitated the bandwidth and performance of FC-switched fabric implementations.

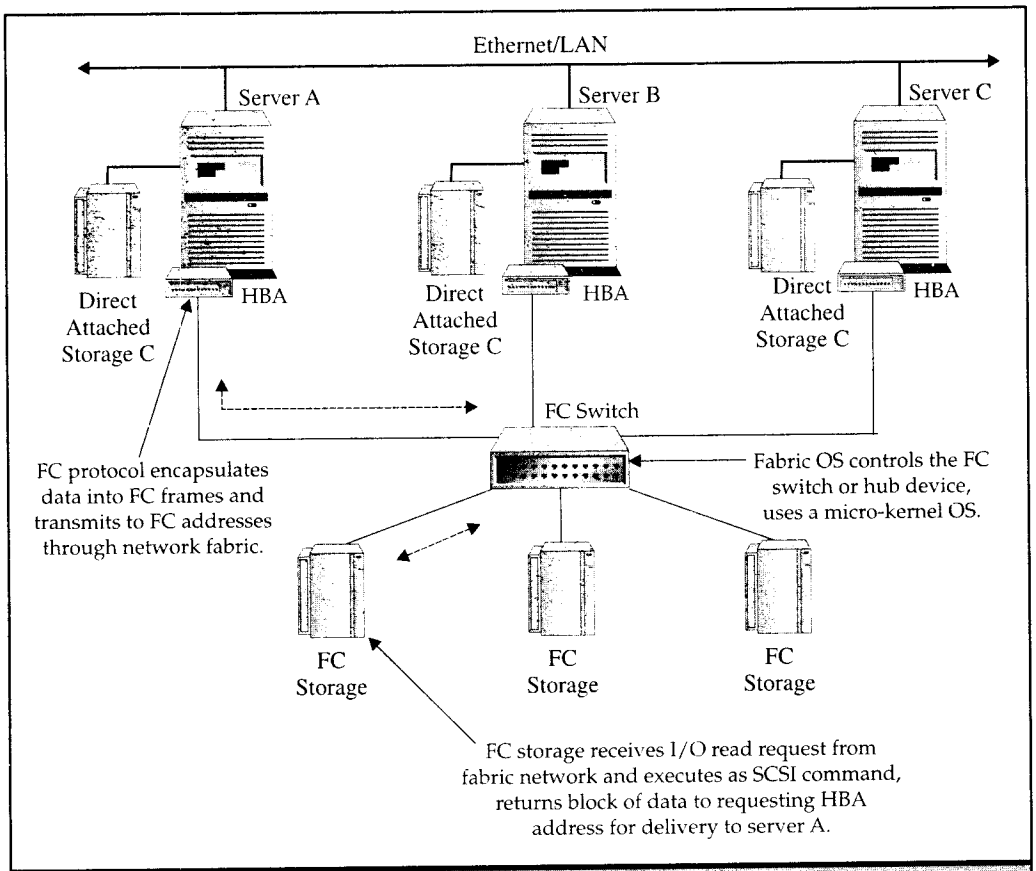


Figure 13-3. *A SAN devices overview and sample read I/O*

The Software Part

The central control of a SAN is contained within the Fabric Operating System functions (sometimes called the SAN OS, or just the fabric). Like micro-kernel components, SAN fabric utilizes the micro-kernel implementation as the basis for its operating system.

The Fabric OS runs within the FC switch. Figure 13-4 shows the typical makeup of the Fabric OS. As we can see, it truly is an effective implementation of a micro-kernel, supporting only the required functions of the fabric. This is good news for performance, but bad news whenever future enhancements and management functions are considered. It has similar limitations to the NAS bundled solution—a micro-kernel conundrum shared among storage networking solutions.

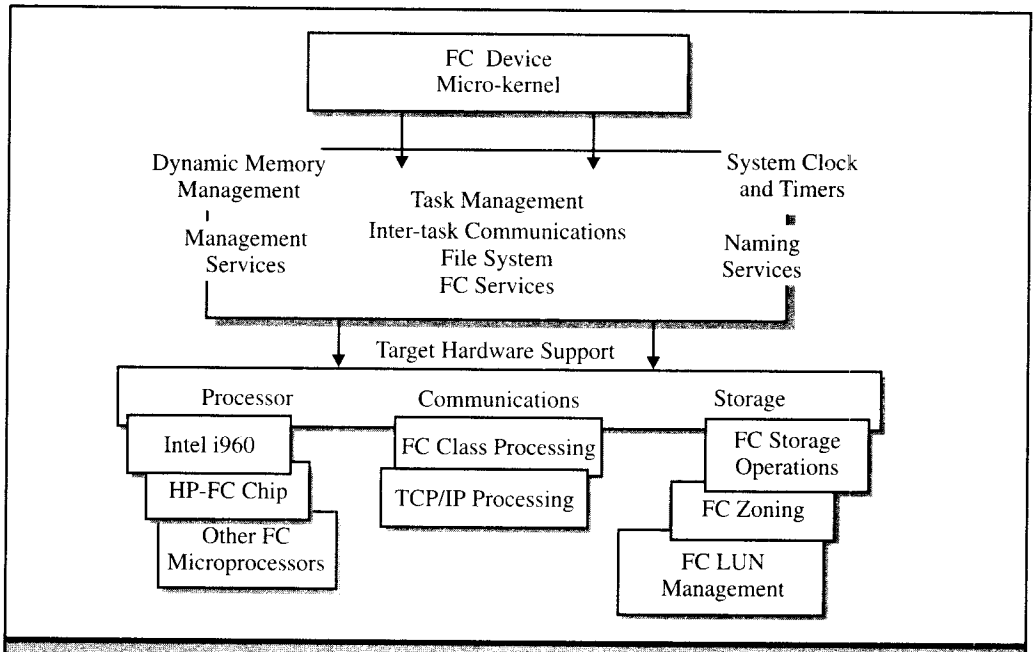


Figure 13-4. *A typical fabric OS micro-kernel*

Fabric OS Services

The Fabric OS offers a common set of services provided by any network. However, it also provides services specific to Fibre Channel and I/O operations. These services are summarized in Figure 13-5 and described next:

- **OS Services** Required functions configured with micro-kernel, such as task, memory, and file management, are included within these services. These form the basic OS functions that provide the core-level processing specific to supporting a fabric.
- **FC Services** These functions interact closely with OS services to facilitate the FC layer processing of frame management within the switch. The FC services provide both switched fabric and arbitrated loop modes of processing, as well as frame processing and protocol routing for the complex interconnections that occur within the switch.
- **Simple Name Services** The FC network works on the basis of a naming convention that identifies devices (such as HBA port and storage adapter port) by their names as they are attached to the fabric. These services provide a rudimentary database (for example, a file that supports the SAN naming conventions, the status of device connections, and so forth) and functions to identify new devices as they are attached.

- **Alias Services** The fabric provides functions that allow particular devices to broadcast frames to a set of aliases as noted within the naming conventions (for instance, a generic set of names that one device can broadcast to).
- **Management Services** FC switches that provide startup and configuration utilities allowing administrators to set up and configure the switch. These utilities range from direct attached access through a PC attached to a serial interface of the switch to web browser-based access if the switch is configured with an Ethernet interface. The management functions over and above configuration and setup consist of status and activity information stored and accessed through a Simple Network Management Protocol (SNMP) database. This database referred to as a MIB (Management Information Base) is specific to SNMP and requires this protocol to extract or store any information. Additional information and discussion of SAN management can be found in Part VI.

Fabric APIs and Applications

FC-switched fabric systems operate similar to other software operating environments by providing a set of basic services to the supporting hardware (refer to the previous bullet points within the OS services discussion). However, like any base operating environment, it must have interfaces to the outside world for access from administrators and other software. These access points will come from application programming interfaces, or APIs. This software interface will provide third-party software vendors, or any systems administrators brave enough to try, the specification for writing

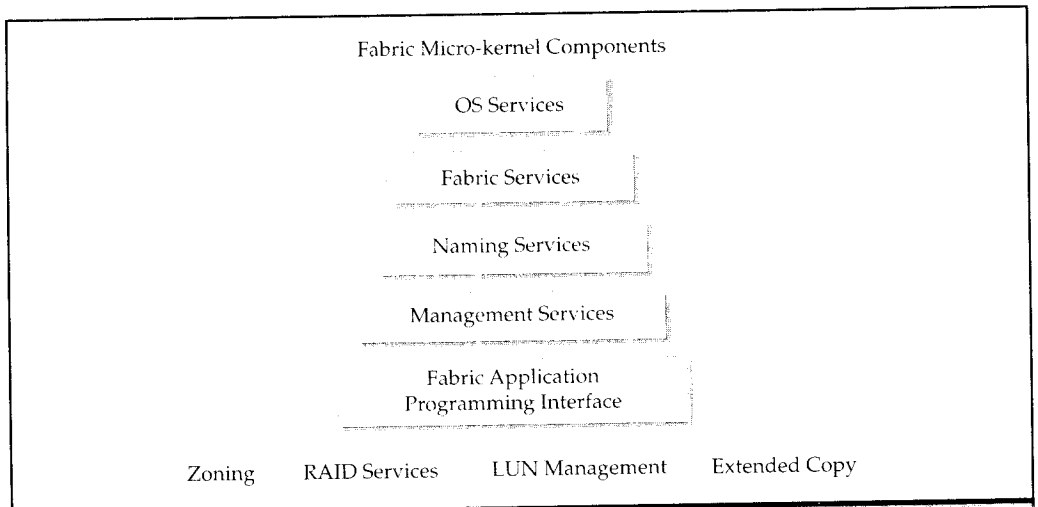


Figure 13-5. *Fabric OS micro-kernel components*

a program to work with the fabric OS. We provide an overview of both API and fabric applications below.

- **Application Programming Interfaces** FC switch vendors all include various levels of Application Programming Interfaces created specifically to allow other software vendors and complementary hardware vendors to develop applications that enhance the operation of the switch. These interfaces are generally so complex that it is beyond most IT administrators to leverage, nor is it recommended by the switch vendors to facilitate local API utilization by customers. Regardless of complexity, this does provide an important feature which enables software vendors—mostly management vendors—to supply applications that interface with the switch and therefore enhance its operation.
- **Fabric Applications** FC fabric supports specific applications that adhere to the vendor's API structure. Applications generally provided by the switch vendor are characterized by their close operation of the switch through facilities such as zoning, hardware enclosure services, and advanced configuration features. Third-party tools, offered as products from software vendors, will interface with the switch to offer mainly management services such as backup and recovery, device and component monitoring, and storage services.

Server OS and HBA Drivers

Dealing with I/Os generated from servers requires that the server operating systems be SAN-compliant. This generally means that the OS is able to recognize the FC Host Bus Adapter driver and related functions necessary to link to SAN attached disks. Although mostly transparent for disk operations, this aspect of the software dependencies and functions supporting the SAN configuration is becoming increasingly complex as file and database management becomes more compliant with SAN network functions.

Most system vendors offering operating systems support SANs. However, it should be noted that further diligence is necessary to identify specific OS release levels that support a specific SAN implementation. Given the variety of hardware and network devices supported through operating system compatible drivers, the support for particular SAN components requires detailed analysis to ensure the attached servers support the SAN infrastructure.

All HBA vendors provide drivers to support various SAN configurations and topologies. These drivers are matched to a specific level of OS. Given most SANs are available as a packaged solution, it is critical that the driver software provides sufficient flexibility to support not only the FC switch configuration, but also the storage arrays and other devices attached, such as routers/bridges, tapes, and other server HBAs. Refer to the OS macro guide to cross reference the SAN support for a particular configuration. As with the server OS consideration, more diligence is necessary to identify specific driver release levels that support a specific SAN implementation and set of hardware components and fabric.

Storage Intelligence

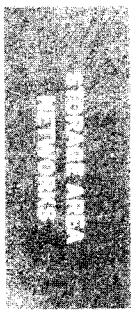
An additional area of software to note is the functional area inherent with particular storage systems. Storage arrays are all delivered with particular implementations of FC, SCSI, and added utility applications. These software implementations must be compatible with both the server OS and fabric OS, as well as the HBA drivers.

Other Networks As a final note on software, we must recognize the effects SAN fabrics are having on TCP/IP and their integration into other enterprise networks. We noted previously that the integration of NAS devices with FC storage components is now available. This configuration does not require a FC switch to support Ethernet attachment. However, it does not develop into a storage area network despite the fact that FC storage can be accessed through TCP/IP stacks and compatible file protocols supported by the NAS device. The capability for a fully functional SAN to participate within the IP network remains problematic.

This also provides a method for switch-to-switch connectivity, thus allowing remote connectivity from one SAN configuration to another. This should be given due consideration, given its ability to execute a block I/O operation through an IP network, something which impacts security, performance, and loading. We examine this in more detail when discussing SAN Connectivity in Chapter 16.

The Hardware Part

The most striking part of a storage area network is its diversity of components. NAS, as we have discussed in Part III, is almost polarized in the opposite direction given its bundled and simplistic structure. The SAN, on the other hand, must have a central network of new devices as well as supporting network appendages attached to the storage and servers it supports. This is most apparent in the required hardware of switches, HBAs, and sometimes bridges and routers. In support of our discussion on the SAN, the following overview will further familarize you with these components.



The Fastest Storage Possible: 100MB/sec

The hardware part of the SAN is made up of components that allow a complete storage network to be enabled. The minimum devices necessary are an FC switch, an FC-enabled server, and FC-enabled storage systems. Although the FC switch could be replaced by a FC hub to facilitate the storage network, we will address it separately (in the section titled "The Unusual Configuration Club, Optical, NAS, and Mainframe Channels" later in the chapter), given its legacy position and its enhancement of switches to handle both FC fabric and arbitrated loop operations.

Figure 13-6 illustrates the minimum components necessary to configure a simple SAN. The FC switch centers the network as it connects the server and storage array. The FC server is connected through an FC Host Bus Adapter (HBA). The FC HBA provides the necessary FC protocol processing and interfaces with the server's operating system. The FC storage array is connected through an integrated FC port attachment that injects the necessary FC protocol communications into the storage controller's mechanisms.

This forms the simplest SAN configuration. However, it is the one that facilitates the FC SAN architecture and provides the basis for additional SAN configurations no matter how complex. In addition to the basic devices, Figure 13-6 also shows the type of connections required for implementation into an existing data center.

The Usual Configuration Gang, RAID, Disk Array, and Tape

Figure 13-7 takes our simple SAN configuration and places into a data center setting. As we can see, the SAN has been enhanced to include additional servers, two storage arrays both supporting RAID levels, and a tape device connected through a FC bridge. This characterizes a sample configuration that supports all the necessary data center operations including database, backup and recovery, and shared access.

The Unusual Configuration Club, Optical, NAS, and Mainframe Channels

Figure 13-8 takes our simple SAN configuration and provides some interesting twists. Although these are unusual implementation scenarios, they are important to point out because they show the increasing levels of flexibility SANs have. Note that there is FC optical storage involved, as well as connection to file storage systems through an IP

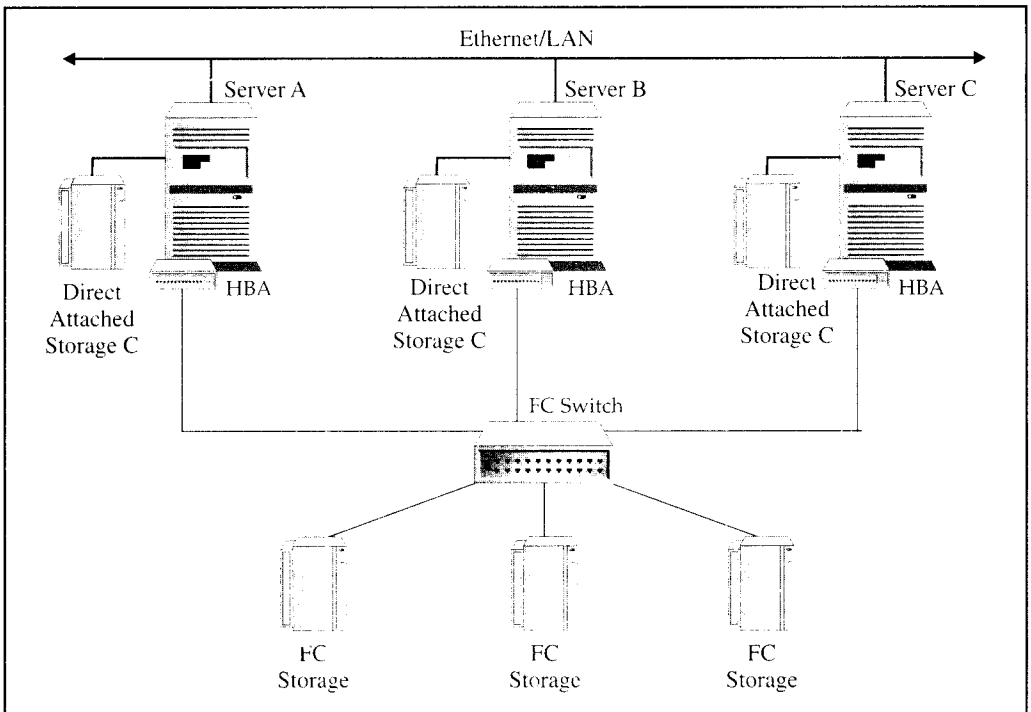


Figure 13-6. *A simple SAN architecture*

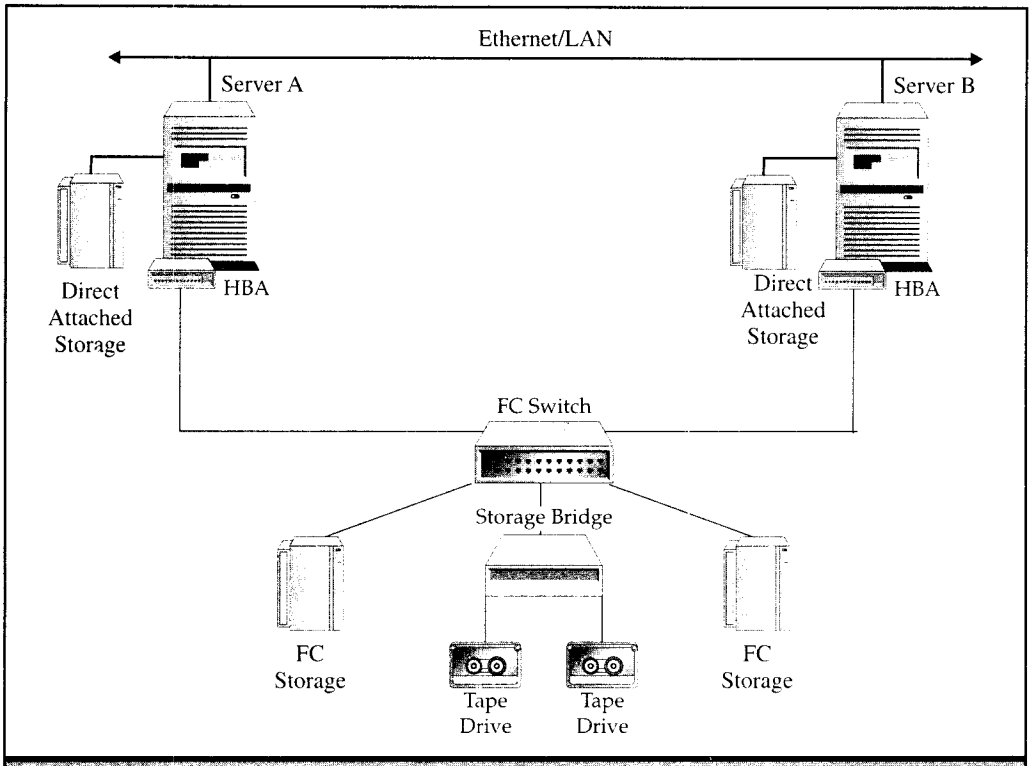


Figure 13-7. *A simple SAN configuration*

connection, and, finally, the connection to alternative servers such as mainframes supporting FC connectivity, like IBM's FICON.

The Connectivity Part

If we further analyze our simple SAN implementation, we find that switch connections come in two forms: the physical connection and the logical port configuration. The physical connection supports both optical and copper connections. Obviously, most switch products operate at optimum using the optical connections; however, the capability to support several connectivity schemes can be important when integrating into existing cabling structures.

The logical connection requires understanding the type of port configuration the switch provides. Ports can be configured as particular types of connections. These are defined according to topology and class of processing. In some switches, these are performed automatically per port, as different types of processing configurations are defined for the fabric—for instance, Classes 1, 4, and 6 for connection types of circuits, and Classes 2 and 3 for connectionless circuits or frame switching (that is, packet-switched networks).

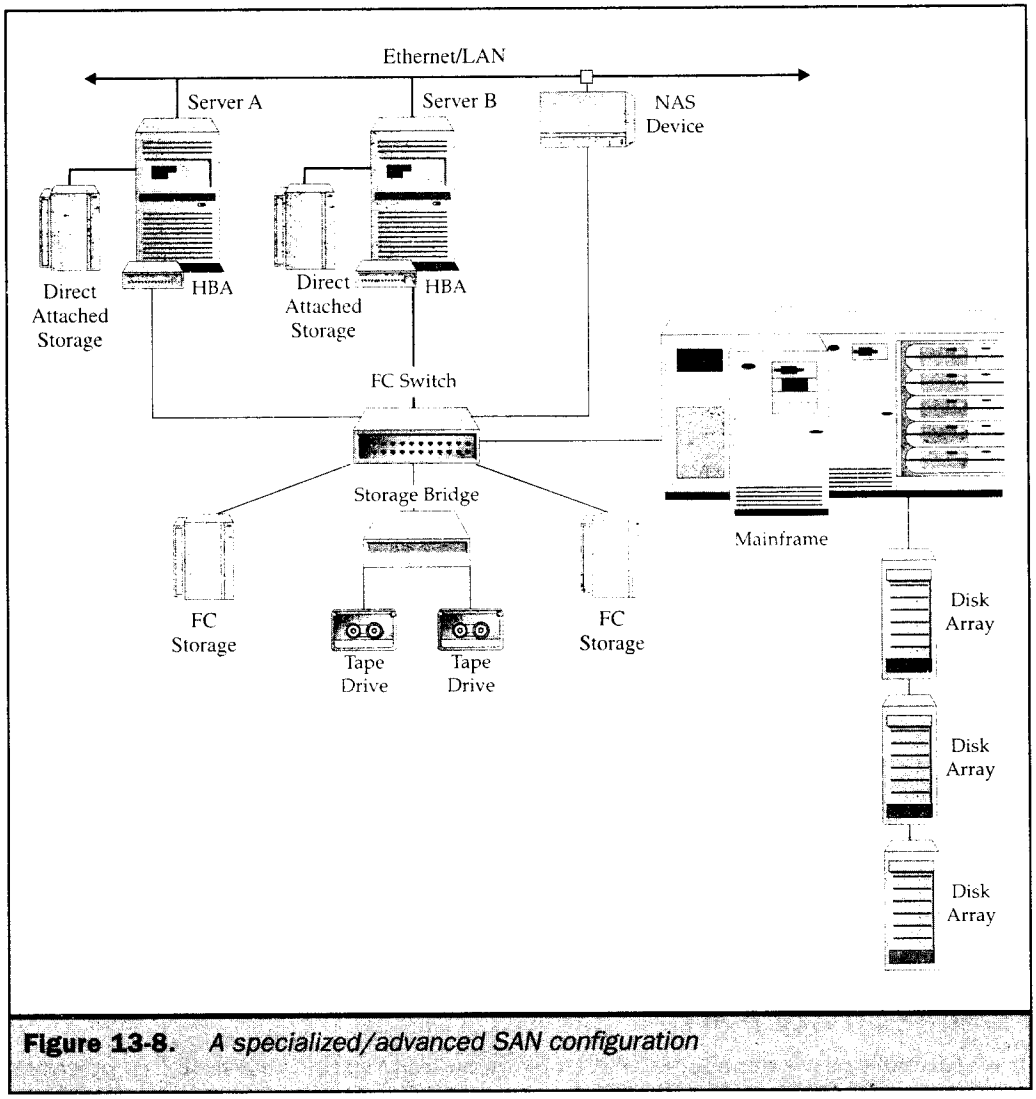


Figure 13-8. *A specialized/advanced SAN configuration*

Connecting the Server: The Host Bus Adapters

Connecting the server requires the use of a Host Bus Adapter (HBA). This is a component similar to the Network Interface Card (NIC) discussed previously in Chapter 8 as network fundamentals in storage connectivity options, and in Chapter 12. It essentially performs the same function as the NIC in terms of providing a physical connection to the FC network, using either copper or optical connectors. In addition, it provides the software functions that translate the FC protocol communications and interfaces these commands with the server operating system. This is provided (similar to NICs) with a set of drivers and is configured according to the type of FC topology the SAN is operating with.

Connecting the Storage: The SCSI/RAID Operations

One of the most effective architectural design strategies within the FC protocol and FC fabric elements is its ability to frame higher-level protocols within communications processing. This means that the SCSI commands that communicate disk and tape operations can continue to operate. These commands are packaged within the Fibre Channel frame and transmitted. Upon delivery, the FC frame is translated and the SCSI commands are reassembled for delivery and execution on the target device.

This is not only effective for server operations, which have historically interfaced with SCSI bus technologies in direct-attached disks, but also for the disk and related controllers themselves. As the SCSI commands are developed from I/O operation activities within the server OS, the I/O operations at the disk controller perform as if they continued to be directly connected. However, this begins to become complex as the reference to logical units (LUN) within a bus are tracked and configured for an FC network. This then becomes a process that grows more challenging when operating with LUNs that are dynamically modified or hidden by layer 2 or 3 processing—say, in the case of a bridge or router.

SAN Configurations

SANs evolve within data centers from simple configurations using single switch or hub operations with a limited number of disk devices. These are often limited traffic projects that provide an entry level of operations and proof of concept for targeted applications. However, they soon grow to configurations with multiple FC switches that are interconnected to optimize the reliability or availability of the applications. The following three levels of configurations best characterize the types of configurations IT will become familiar with.

Entry Level

An entry-level configuration will basically be a proof of concept or beta installation to prove the reliability of the SAN project. As stated earlier, these are often limited projects handling limited user traffic and designed to support a targeted single application. The entry-level SAN configuration can also allow IT personnel to become familiar and comfortable with the new types of hardware, give them some experience with the new software fabric configurations and operation, and let them experience real traffic within the SAN.

Figure 13-9 shows an example configuration of an entry-level SAN. Note this configuration contains a single FC switch, two storage arrays, and a tape device. There are three Windows-based servers attached to the switch which complete the SAN network. In this example, anywhere from three to five relational database systems provide production test beds for application testing. Each database has its aggregate data stored across the storage arrays. The storage arrays themselves are configured for RAID level 5 and level 1. This allows several databases to share storage in a single array; in this case, array A using RAID level 5, while array B is configured for RAID level 1.

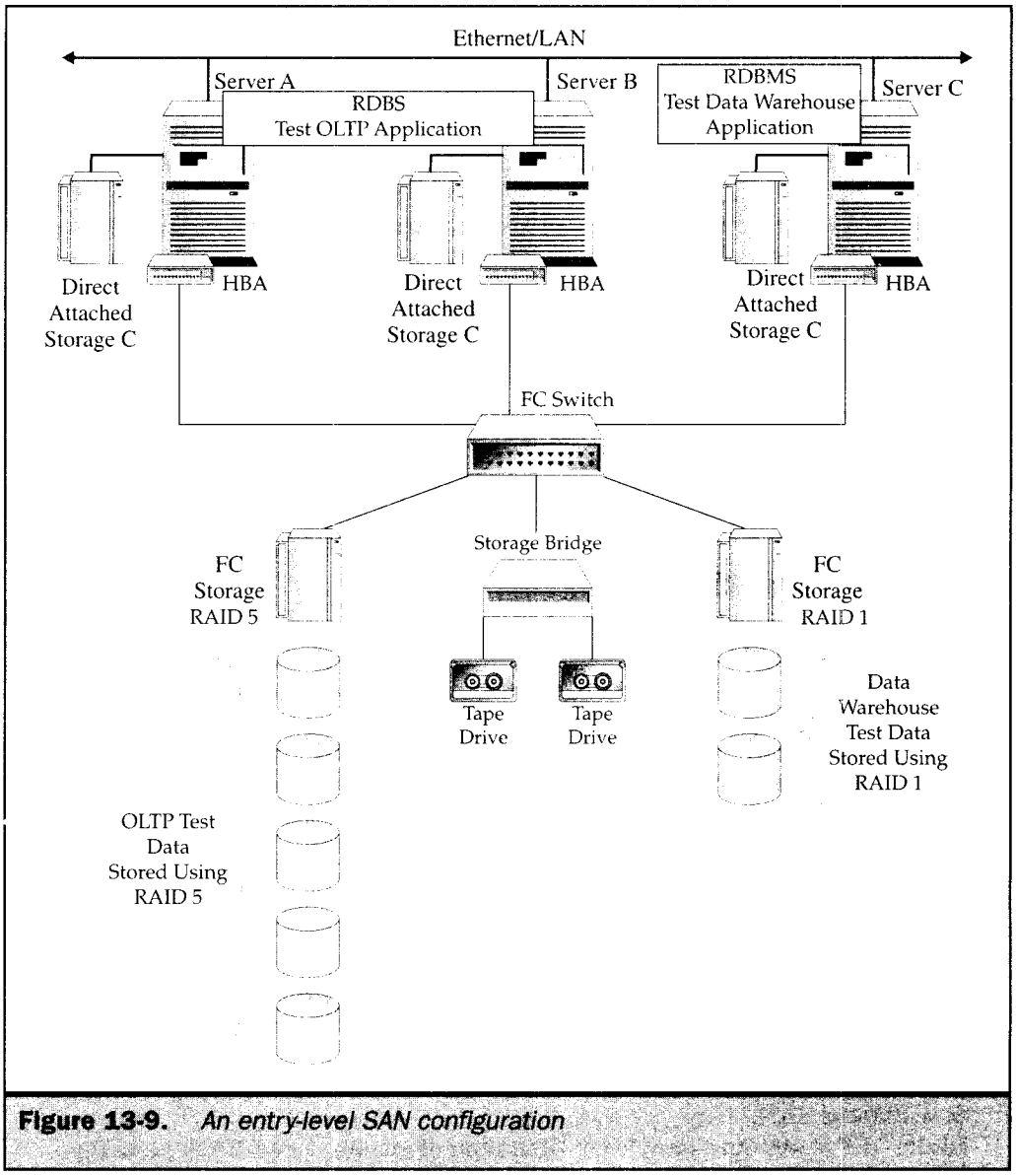


Figure 13-9. *An entry-level SAN configuration*

This configuration provides an entry-level SAN that provides a test bed for three database applications; some are for typical OLTP workloads while the third is for a data warehouse application. The systems are set to allow applications and DBAs to test new query applications against test databases. The SAN configuration replaces the six servers used to support database testing and the six storage arrays directly attached to their respective servers.

Consequently, even for an entry configuration, the architecture of the SAN can consolidate a considerable amount of hardware and software.

Mid-Range: Production Workloads, Multiple Switches

If we move up to supporting production workloads, we must consider all the necessary reliability and availability configuration requirements. Figure 13-10 shows an example SAN that supports production applications with ten servers using Windows OSs. The servers are attached to three FC switches configured in what's called a *cascading arrangement*. The multiple storage arrays support both RAID 5 and RAID 0 in our example set of applications. Our production configuration also has the addition of backup with the attachment of a tape library.

Our production workload supports several database applications that utilize five of the ten servers. These transactional OLTP-type requirements utilize the storage arrays configured for RAID 5. Our other workloads are a combination of an intranet for internal users of the company's internal web application. This application utilizes the storage arrays configured for RAID 0, which has the data stripped but has no failover capability.

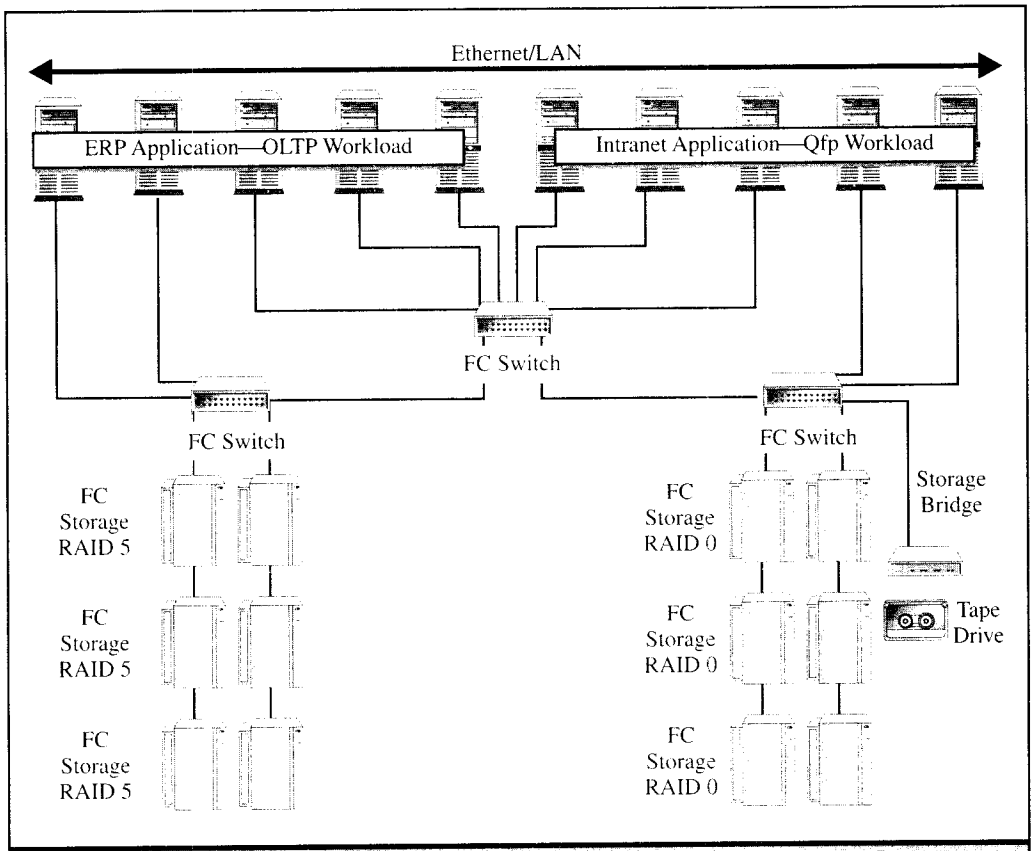


Figure 13-10. *A mid-range SAN configuration*

Enterprise: Director-Level with Duplicate Paths and Redundancy

Moving up to the support of an enterprise level and the support of multiple applications, we have an example enterprise configuration in Figure 13-11. This much larger configuration shows 20 servers, consisting of both UNIX- and Windows-based OSs, attached to six FC switches. Storage arrays support 2 terabytes of both file- and database-oriented data models, given that the workloads range from OLTP database applications to web servers, and data-centric datamart applications. A datamart, incidentally, is similar to a data warehouse with limited subject scope; in some industry sectors, however, datamarts have proliferated faster given their singular and more simplistic database design. This allows for faster design and implementation activities, which makes them demanding in their storage capacity, access and data sourcing requirements from larger data warehouses, and operational databases.

The storage arrays are configured in RAID levels 5, 1, and 0. As with any production workloads, appropriate failover has been configured with a more complex cascading switch configuration. There is a new twist, however—a remote connection within the switch that allows for a disaster recovery operation to be in place. The usual configuration of a tape library to facilitate the backup strategies of various workloads and applications is also in place.

As demonstrated here, SAN configurations have an inherent value to storage network architecture. Even though the architecture reflects a new model of processing, SAN configurations provide the most effective and efficient method of server consolidation, storage consolidation, improved data access, and scalability of data storage.

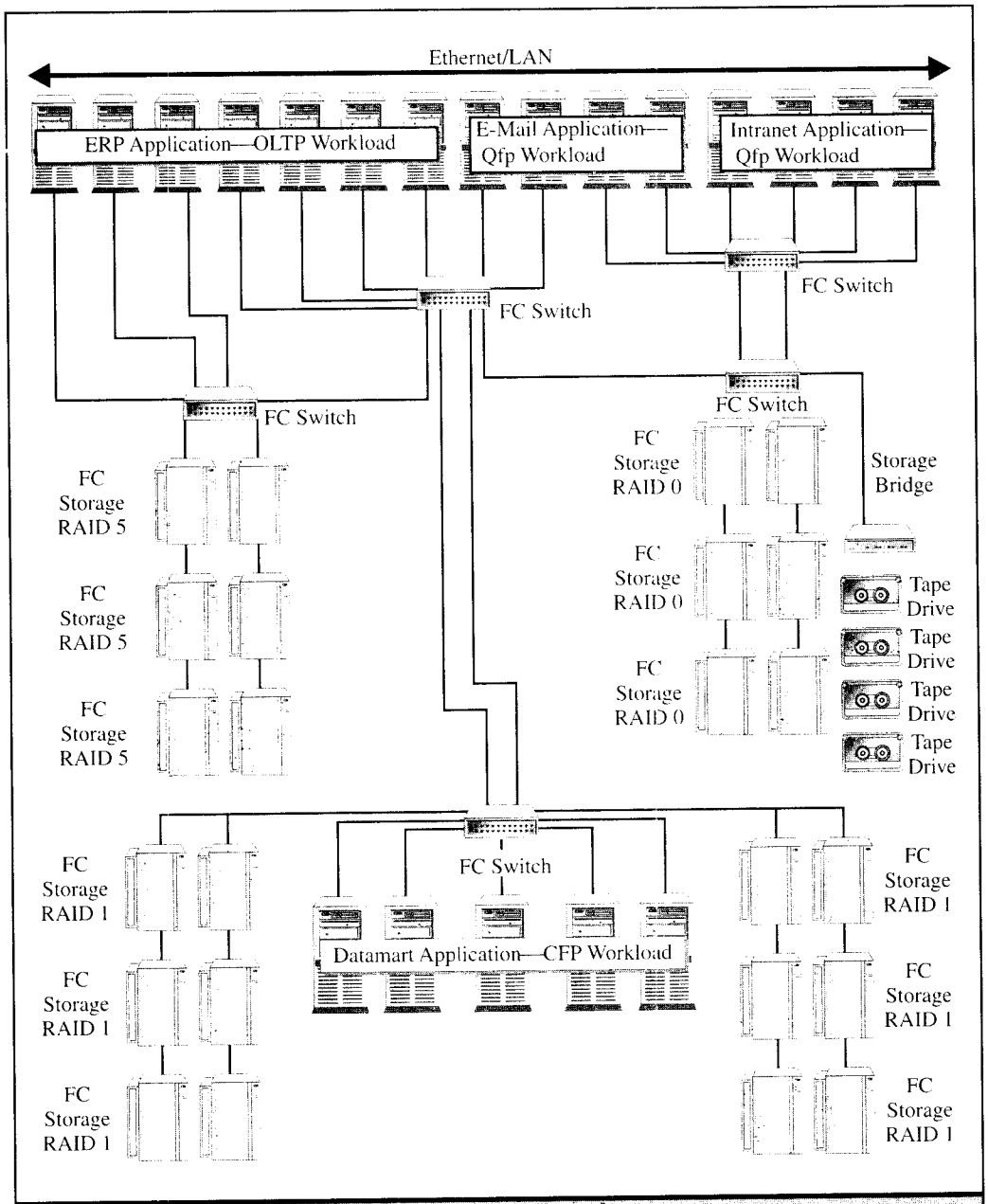


Figure 13-11. *An enterprise SAN configuration*

The Complete Reference

Chapter 14

Hardware Devices

SAN hardware is divided into three distinct parts: the Fibre Channel switch, the Host Bus Adapter (HBA), and the storage device itself. In addition, one must consider bridging, or routing, solutions. Chapter 13 is the continuation of the detailed discussion of SANs started in Chapter 12, which covered their specific architecture. This chapter will explore the hardware components that make up the SAN, a summary of what they are and how they operate, and a view of fabric operation from a hardware perspective.

The switch part is a discussion of the Fibre Channel (FC) switch component that essentially becomes the heart of the SAN. Included will be a discussion of the device attachment options, a brief explanation of operation, and the role they play in the configuration of the switch, consisting of the foundations for configuration, port identification, and interswitch links.

The FC Host Bus Adapter (HBA) is a necessary component to attach servers to a SAN. HBAs have ports that connect to the switch, which play a pivotal role in communicating with the SAN fabric on behalf of the server. There are also considerable differences in HBA software drivers and OS support, as well as options for HBA functions and their interaction with FC switch.

Perhaps their most important aspect, given that SANs are "storage" networks, is the FC-enabled storage itself. Storage devices for SANs require that they communicate using the Fibre Channel networking standard, consequently their roles and various operations differ in comparison to direct attached storage devices. Included with this discussion will be an orientation to tape media into the SAN, where we identify an additional supporting component, the FC bridge/router, which is necessary in developing a fully functional FC-based storage area network.

We will conclude the chapter with a brief discussion on several important points in the operation of the SAN fabric. From a hardware perspective, the modes of processing and the functions of the physical ports become required knowledge when designing and implementing the SAN. Given that the SAN's heart is the FC switch, the operational aspects of how the switch provides a circulation of data throughout its system cannot be overlooked. SAN switches will be used in multiple configurations (covered in detail in Chapter 18). This requires an initial understanding of the switch-linking functions used for expansion, redundancy, and recovery configurations. Discussions of such can be found in this chapter, as well as in Chapter 15.

The Fibre Channel Switch

The Fibre Channel switch is the heart of any Storage Area Network configuration. It connects the user to the data being stored in the SAN. Like a heart, all information traveling through this closed system will sooner or later pump through the switch. Within the switch are connections, called ports, which function in different ways depending on the type of device connected. There is an effective Fibre Channel standard, called the T11.3 standard, which is the ANSI standard governed by the X3T11 group and defines the usage of the port via the Fibre Channel protocol. The T11.3 standard

provides a set of guidelines for vendors to follow—a common language, so to speak. The Fibre Channel standard defines three types of ports:

- **The F_Port** Used to attach nodes. In storage networking, nodes refer to a network logical term for attached devices (for example, servers, storage devices, routers, and bridges, which are attached to the switch).
- **The E_Port** Used to connect one switch to another switch.
- **The G_Port** A generic port that can be pressed into service as an F_Port or an E_Port depending on the vendor's implementation.

It should be noted that, in general, Fibre Channel switches are categorized in terms of port capacity and their capability to sustain port failures. Low-end switches, while cost-effective, support only a base set of functionality in connecting devices to the network; moreover, the ability to connect to other switches or fabrics is minimal. Low-end switches offer no real port failure tolerance. The class of switches defined for entry level and mid-range usage is referred to by the generic term *Fibre Channel switch*. A glass ceiling of port counts exists that separates the FC SAN switch from the *Director* level products. With port counts at or above 64, including internal path redundancies for fault tolerance, the Director class storage switches are used for large enterprise class storage configurations. Feature function notwithstanding, there is a clear cost difference between low-end Fibre Channel switches and Director Class switches. The difference is not only clear, it is quite large.

Unless a user has a high-end requirement, most users start with a handful of low-end switches, as well as the myriad problems associated with them, and end up very quickly needing to bump up to Director Class switches. The point is, you may not need the ports now, but you may very well need the functionality depending on the service level requirements of the applications being supported.

The F_Port

As stated previously, the F_Port, also referred to as a Fabric Port, connects server and storage devices to the switch itself. A device plugged into the switch's F_Port is referred to as a *node*, and, in FC terms, is identified as an N_Port. If used in an arbitrated loop topology, it becomes an NL_Port. For instance, a server, when plugged into the switch, creates an N_Port, whereas a storage device using an arbitrated loop is recognized as an NL_Port. A basic F_Port is shown in Figure 14-1. Operating as FC nodes, they are identified by the switch as their particular N_Port or NL_Port designation.

All devices attached to a Fibre Channel switch must log in. When a device is attached, it accesses a file in the Name Server database within the switch that contains information explaining to the server just what this device is. The Name Server informs the switch of the device's name, address, type, and class of service. It is the formal introduction between switch and device. Anyone putting together a Storage Area Network must make sure the devices they are planning to attach (or are going to attach in the future) are supported by the switch. If a device is not supported, it becomes all

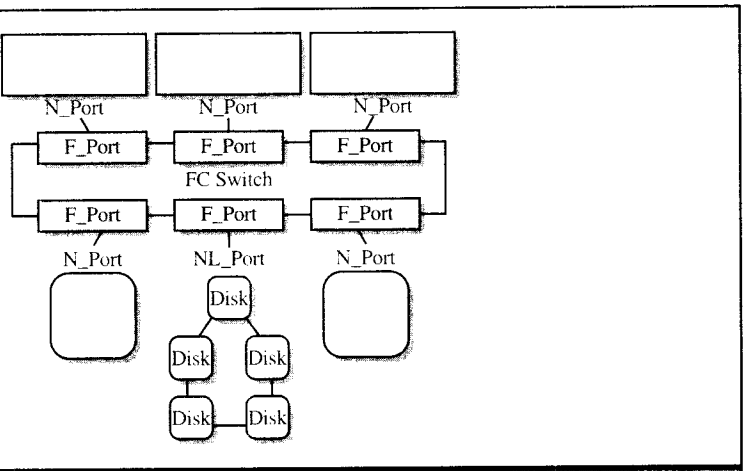


Figure 14-1. *Port configurations*

but useless to the SAN. Keep in mind, too, that arbitrated loop login will take longer than general fabric login or import login, given the extra loop and device assignments.

The Name Server, shown in Figure 14-2, registers a device's port address, port name, node name, class of service parameters, and the Fibre Channel layer 4 (FC-4) protocols it supports, as well as the type of port. The Name Server also provides the information one device, like a server, needs to find another device on the network, like the storage. Port name and port address are contained within a single table so that a device can be referred to by its name or by its address, much the way a web site can be called up by its URL or IP address. This is helpful as Storage Area Networks increasingly have IP ports made available to them. The Name Server supplies a method of recognition between the two types of ports, regardless of the protocol. When connecting switches, it is paramount that each switch's Name Server be synchronized and consistent with one another.

Any changes made to the devices attached to the switch are handled by either voluntary or involuntary services provided within the switch. As devices or connected switches are upgraded, port usage can become volatile. This is an important aspect in change management within the port itself. How these changes are made is dependent upon vendor service and implementation. Certain vendors automatically update the Name Server using State Change Notification (SCN) or Registered State Change Notification, which is an example of an involuntary change. A voluntary service consists of, you guessed it, using the two preceding methods to update the Name Server yourself. At this point, you may be asking yourself—and rightly so—just how is that voluntary?

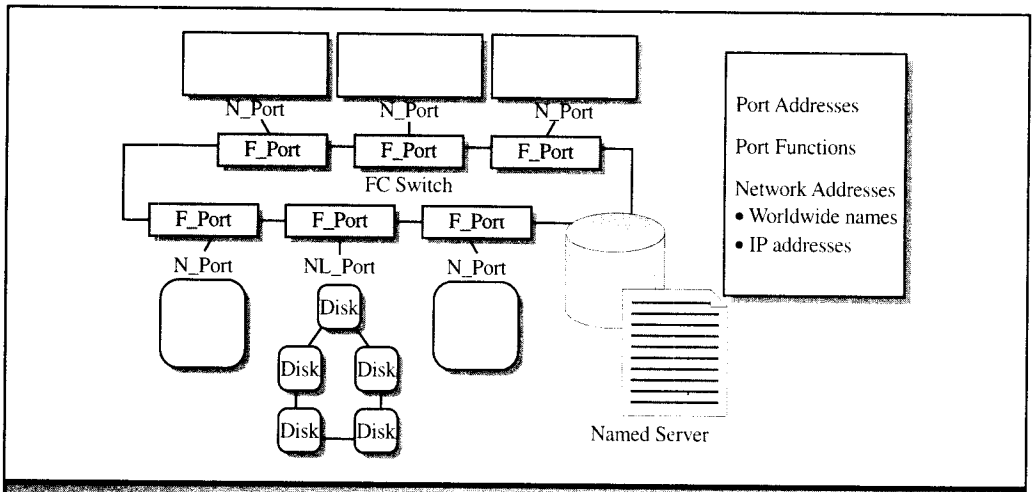


Figure 14-2. *The name server contains names, addresses, and information on nodes.*

The E_Port

E_Ports, also referred to as expansion ports, connect one switch to another. These switch ports have become critical given the rapid expansion of multiple switch configurations needed to support larger and larger server and storage capacities. E_Ports are now fundamental to the operation of a Storage Area Network and should be considered carefully when designing and implementing any SAN configuration.

The E_Port, as shown in Figure 14-3, provides the means by which one switch fabric communicates with another switch fabric. For this to happen, a compatible connection is required. This allows the E_Port to successfully coordinate with the Name Servers within each switch, synchronize frame transfers between switches, and facilitate interswitch access to storage resources. Utilizing arbitrated loop configurations between switches, while possible, increases the overall risk factor in the efficiency of an E_Port because of the increased overhead and treatment of the NL port as a separate network, which dramatically increases the number of addressable units and decreases bandwidth utilization.

The G_Port

The G_Port is a generic port and, depending on the vendor's implementation, can be used as either an F_Port or an E_Port. In other words, G_Ports are universal ports that can be used for a combination of functions. Although we haven't covered the necessary overhead in implementing a port configuration for issues such as recoverability and

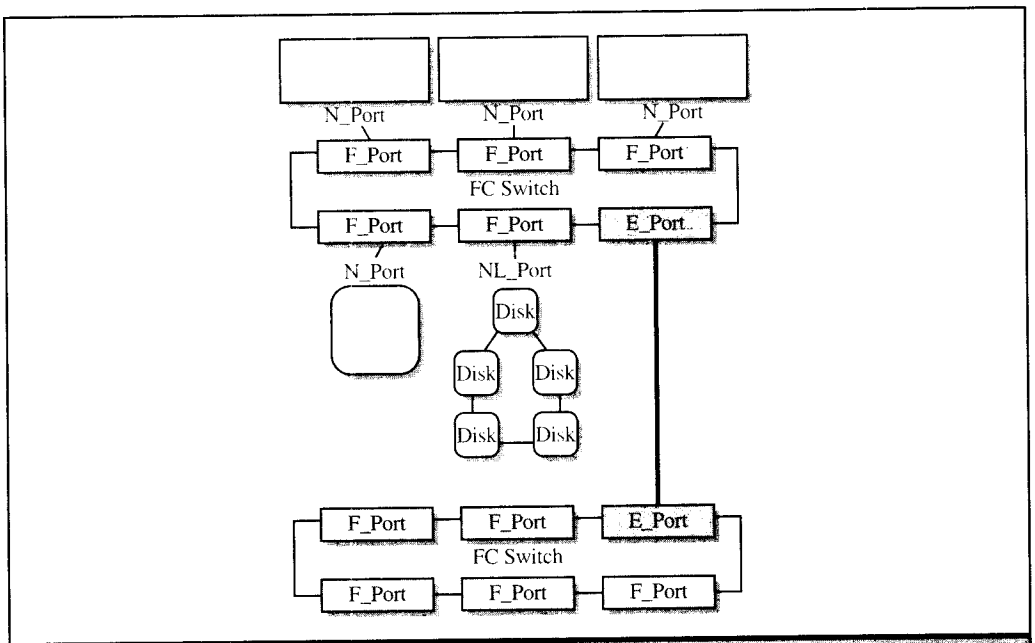


Figure 14-3. *Extending the switch with an E_Port*

redundancy, G_Ports are likely candidates for recoverability and redundancy usage in light of port failure.

G_Port populations will dramatically increase, as ports become multifunctional and autosensing becomes more prevalent. Switches will come equipped with 64 so-called generic ports, and will distinguish themselves as either F_Ports or E_Ports when a device or an additional switch is attached. This adds critical switch flexibility when configuring multiple switch operations—all of which increases the cost per port.

Host Bus Adaptors

Simply put, the Host Bus Adapter (HBA) is the link between the server and the Storage Area Network. Similar to the Network Interface Card (NIC), HBAs provide the translation between server protocol and switch protocol. HBAs connect to a server's PCI bus (see Chapter 7) and come with software drivers (discussed in greater detail in the following chapter) that support fabric topologies and arbitrated loop configurations.

HBAs are available in single port or multiple port configurations. Multiple ports allow additional data paths for workloads moving between the server and the switch via a single HBA. In addition to containing multiple ports (a maximum of four, at this point), a single server can hold, at most, four HBAs. Today, any single server can possess 16 ports (four ports times four HBAs), or 16 separate points of entry into the switch. However, keep in mind that four discrete ports on one HBA increases the risk for single point of failure along those data paths.

In providing the initial communication with the I/O from the server, HBAs encapsulate SCSI disk commands into the Fibre Channel layer 2 processing. HBAs communicate within the FC standard through the class of service defined by the Name Server at the time of login. As such, the HBA plays a key role in providing levels of efficiency in executing the operating system's I/O operations. Figure 14-4 illustrates an HBA's basic functions.

One key duty of any HBA worth its bandwidth is discovering and mapping the storage resources available to it within the switch fabric. This mapping is critical to the devices that are available to any particular server. As will be discussed in the next chapter, there are various ways to restrict access to storage resources, such as zoning. It is important to note here, though, that HBAs must deal with this issue in an effort to understand the devices it must contact on behalf of the server, which it ultimately works for.

As you might expect, all of this becomes even more complex in supporting arbitrated loop devices on the switch.

Depending on the vendor, additional functionality is bundled with different HBAs. These functions range from software RAID functionality to advanced management functions (for example, diagnostic functions and the new enclosure services provided by many vendors). In the future, additional functionality will come to include a virtual interface that bypasses much of the layered processing currently required. For more

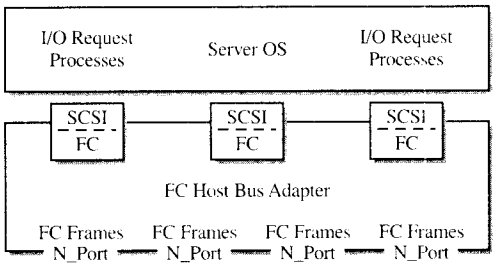
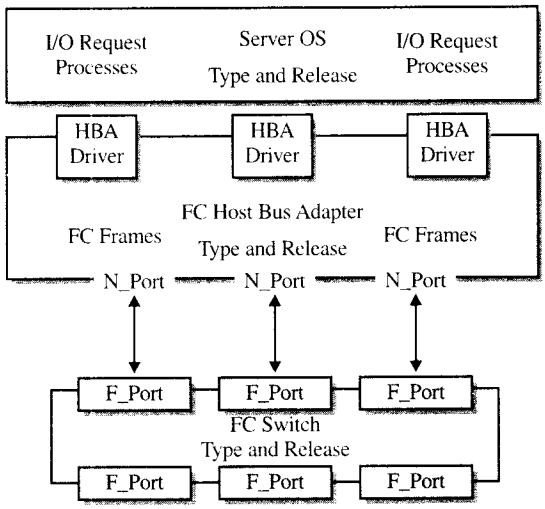


Figure 14-4. *The basic functions of the FC Host Bus Adapter*

information on this, check out the discussion on InfiniBand in Chapter 20. But don't say I didn't warn you.

The major reliability questions of any HBA are twofold. First, the HBA's compatibility with its server's operating system is key to the effective operation of the FC network in total because each operating environment has unique differences in how it handles base I/O, file system, and buffering/caching methods. It is important to understand at a macro-level the differences between a UNIX integration and implementation of an HBA versus integration and implementation within a Windows environment. The second factor is the compatibility, at a software level, of the switch's fabric operating system and the HBA's software drivers. This requires an understanding of the supported release levels of any given switch vendor against any particular HBA vendor. Since many SAN components are acquired through OEM relationships, compatibility can become sticky when intermixing equipment from different vendors. As shown in Figure 14-5, HBAs play a critical role in the interoperability of a SAN configuration.



Server OS, FC HBA, and FC switch release levers must all be compatible to match operating systems, installed storage types, and switch port functions. The critical link between the I/O request and storage network is the HBA.

Figure 14-5. *HBA's critical role in interoperability*

Putting the Storage in Storage Area Networks

Given that SANs are *Storage* Area Networks, perhaps the single most exigent component in the configuration is the storage itself. SANs require storage that is Fibre Channel-enabled. In other words, FC disks must be connected to the switch via an FC topology. Fibre Channel-enabled disks come in two configurations: JBOD, Just a Bunch Of Disks, and RAID, Redundant Array of Independent Disks. Each configuration maintains its own set of FC features and implementations.

FC storage arrays, which are multiple disks hooked together, are connected in a loop configuration, as depicted in Figure 14-6. Loop configuration allows each disk to be addressed as its own unique entity inside the segment. FC storage arrays (depicted in Figure 14-6 as well) also provide additional functions, such as the stripping of data across the units, which allows a single outsized file to be spread across two, three, or even four drives in the array. In addition, most provide low-level enclosure management software that monitors the device's physical attributes (its temperature, voltage, fan operation, and so on).

JBOD

A typical JBOD configuration is connected to the switch through an NL port. Most implementations provide dual loop capacity whereby redundancy protects against

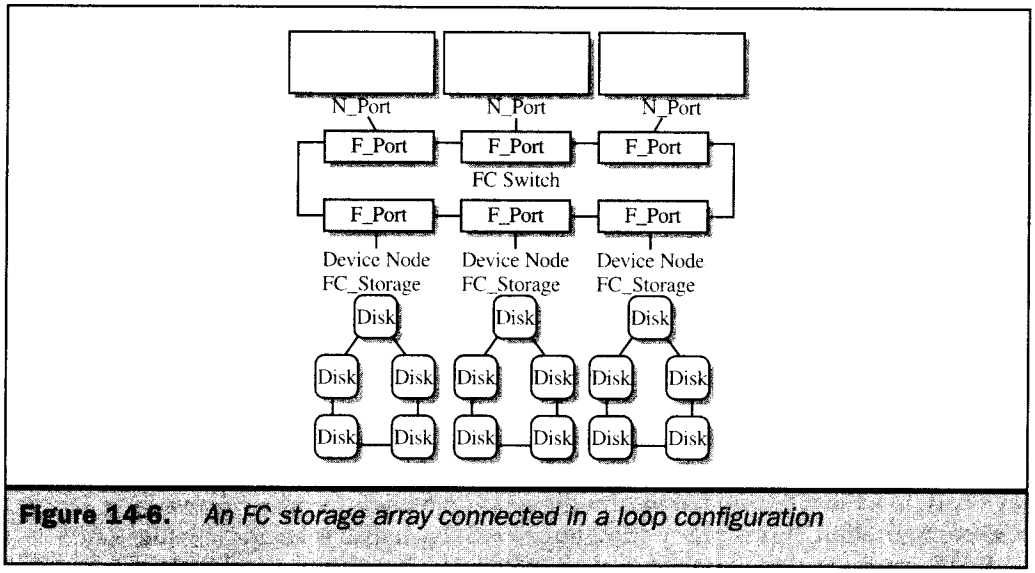


Figure 14-6. *An FC storage array connected in a loop configuration*

single loop failure. In other words, should one loop go down, the information on the storage device can be retrieved via the second loop. A dual loop requires four of the switch's ports. Another, less typical method of attaching a JBOD array to a switch is to split the devices into separate loops. A JBOD array of eight drives could have one loop serving drives 1–4 and a second loop for drives 5–8. This method also requires four ports on the switch. The benefits of splitting the devices into several loops include shorter path lengths and less arbitration overhead within the loop itself.

The disadvantage of any JBOD implementation is its lack of fault resiliency, though there are software RAID products that allow the stripping of data with recovery mechanisms encompassed in the JBOD enclosure. Given the number of disks and the transmission of SCSI commands to multiple targets, using RAID software in conjunction with a JBOD implementation presents its own problems. It is important to keep in mind that while these are Fibre Channel-enabled disks, the disk drives themselves execute a SCSI command set when performing read/writes.

RAID

A Fibre Channel-enabled RAID storage array places a controller in front of the disk array that provides and controls the RAID level for the disk array (RAID 1–5). RAID offers a way of protecting the data by creating an array of disk drives that are viewed as one logical volume. Inside this logical volume, which may consist of seven drives, data is partitioned, as is recovery information. In the event of a single drive failure, the array reassembles the information on the remaining drives and continues to run. The RAID controller uses either an N_Port or a NL_Port depending on how the vendor put the disk enclosure together.

Because the RAID controller stands in front of the array, the FC-enabled disks, regardless of their configuration, become transparent. The switch only sees one device, not several linked together. A single RAID controller can also control several arrays, as indicated in Figure 14-7. For example, four volumes, each volume containing seven disks, would equal a RAID system of 28 disks. Still, the switch only sees the RAID controller—just one device. In this scenario, more often than not, the RAID controller will utilize an N_Port, not a loop.

The key advantage of Fibre Channel RAID is its ability to provide levels of fault resistance for hardware failures, a feature not found in JBOD configurations. For enterprise-level workloads, this feature is all but mandatory.

Just as the HBA is the critical point between a server's operating system and the switch's operating system, RAID controllers require a level of specific Fibre Channel software that must be compatible with the switch and the HBA. As noted previously, it is the HBA's job to inform the server which disks are available. In a JBOD configuration, this is pretty straightforward. Each disk is an addressable unit. In a RAID configuration, it becomes the controller's duty to specify which disk is addressable. The RAID controller, via the software contained within it, has to identify itself to the switch, specifically the Name Server within the switch, as well as the HBA,

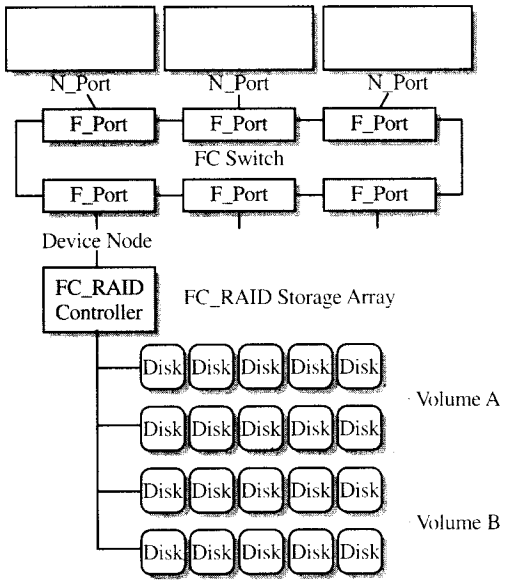


Figure 14-7. *An FC RAID storage configuration*

in order for the server to know which disks it has access to on the network. This can quickly become confusing, given that RAID deals in logical units, not independent addressable disks.

One last word about Fibre Channel storage, and this goes for RAID and JBOD configurations: when assigning multiple servers to the switch (via the HBA), the servers have to be told which storage resources they are allowed to play with. And this can quickly become tedious. For example, each server has its own file system, and that file system must reflect the location of the files the server has access to. Problems arise when two or more servers have access to the same files. What happens when two servers reach out for the same file? You guessed it... trouble, headaches, and a whole lot of shouting. File sharing between servers attached to a Storage Area Network remains a tedious and problematic issue. Consequently, zoning and masking each server's authorized resources continues to be a prerequisite for effective operation. Before *you* start shouting, and reach for the aspirin, have a look at Chapter 22.

Bridges and Routers

In addition to the challenges posed by file sharing, data sharing, and device sharing, there are standard data center practices that are required for any type of storage model. Every data center must protect the data stored within the arrays. Most accomplish this using backup/recovery software and practices, which entails the use of tape devices as

the primary media for archival copy and data copy functions. In a SAN environment, however, this is easier said than done. Fibre Channel-enabled tape devices have been problematic in their support of this new storage model.

To overcome this hurdle, a new type of device was required to bridge the FC protocol into a SCSI bus architecture used in tape media. Because of tape technologies' sequential nature and the resulting complexity entailed with error recovery, tape media has been difficult to integrate. Solving these difficulties required a device that not only bridged the Fibre Channel into the tape controller/drive bus system, but also further required the management of the logical unit numbers (LUNs) that were utilized in the tape's SCSI configuration. The solution was found in bridges, or routers, for Fibre Channel. This is illustrated in Figure 14-8.

Although they are compatible with other SCSI devices, routers are primarily known for their capability to facilitate the operation of tape media within a SAN. Routers provide an effective means of establishing a tape media library for SAN configurations. The alternative would be to copy data from the FC storage arrays onto the LAN and shoot it off to a backup server with a directly attached SCSI tape drive. Considering the

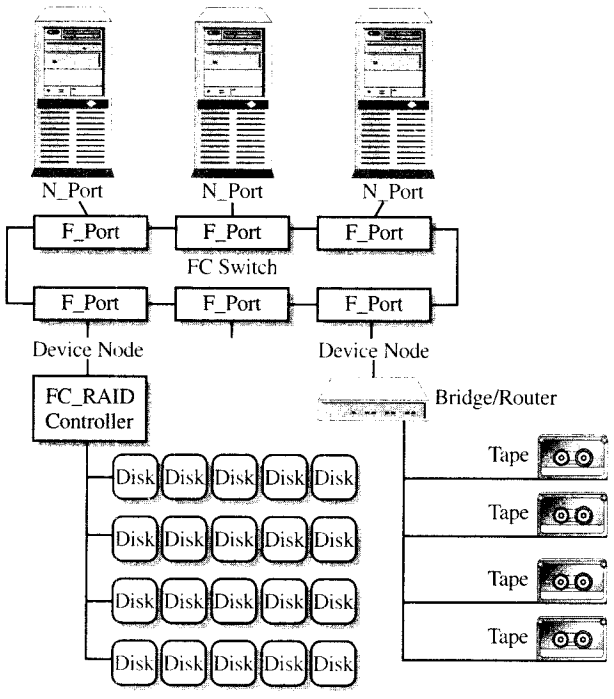


Figure 14-8. *A simple bridge/router configuration*

overhead required, routers provide a much sleeker configuration for data protection. As always though, even the sleek solutions have their drawbacks. Fibre Channel to SCSI routers bring their own performance issues in matching the switch's port performance with the SCSI bus attachment at the other end of the bridge. Due to SCSI transfer rates, speeds across the router will be constantly slower and throughput will be compromised.

When integrating a router solution, it is important to understand what is needed, whether it's a discrete component or one of the integrated solutions that are increasingly creeping into tape subsystem products. In looking at either of these solutions, you'll have to distinguish other protocols besides SCSI that may be necessary to bridge into your SAN configuration. Router technology is beginning to move toward a gateway type solution where Fibre Channel integrates additional I/O and network protocols.

An additional word of caution: routers add yet another level of microkernel operating environment that must be compatible with all of the components across the Storage Area Network. Routers must also be compatible with a significant number of tape systems, which only adds to the complexity of implementation. The surest bet is to approach the integrated solution supported by the tape manufacturer.

Fabric Operation from a Hardware Perspective

The SAN's fabric operates through, and is governed by, the logical components of the Fibre Channel standard previously mentioned. The standard is broken down into logical constructs of frame, sequence, and exchanges. FC provides its storage flexibility through the ability to transport other protocols, such as the SCSI protocol.

- **Frame** The logical construct of transporting data through the switch.
- **Sequence** A block of numbered or related frames transported in sequence from initiator to target. Error recovery takes place as the receiving port processes the sequence.
- **Exchange** A number of nonconcurrent sequences processed bidirectionally (it should be noted, however, that a port can only process one sequence at a time). Although a single port is limited to single sequence processing, it can process multiple simultaneous exchanges.

Within the overall FC protocol standard are embedded the primitive sequence protocols used for error recovery during exchange processing. Fabric login is used to communicate the port's operation characteristics. Operations are directed by the N_Port login and logout commands, which provide session-level parameters as ports issue and exchange sequences. The exchanges are made up of frames (as shown in Figure 14-9) and indicate the large user data area or payload that allow FC protocols to transmit at gigabit speeds.

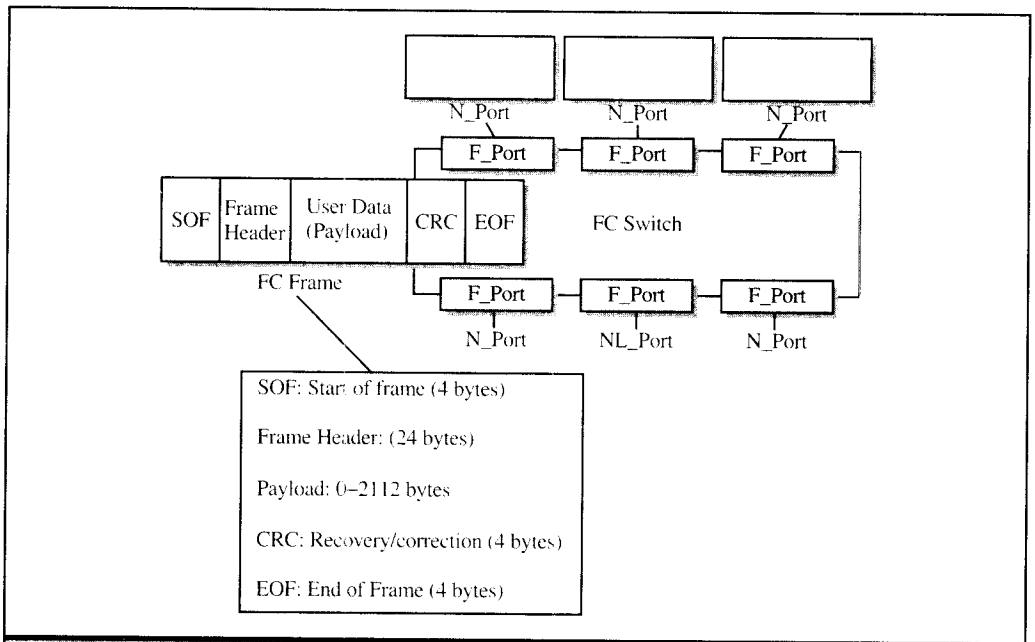


Figure 14-9. *Anatomy of the FC Frame*

Selected by the port at login, the class of service and other parameters the switch will support directs flow control of the frames. Flow control is determined by connection attributes between communicating devices. This process is similar to network operations where connections are either mindful of their state and acknowledge transmission and receipt, or disregard their state, acknowledgment, and effective frame recovery. Although there are six classes of operation defined by the FC standard, only three are useful to data storage operations, and therefore worthy of mention.

- **Class 1** A dedicated connection in which two devices establish complete control and dedicated bandwidth of the link. Until they complete their transactions, the ports remain busy and unavailable to any other traffic. Although useful for batch type operations and data streaming when the complete bandwidth of the link can or must be used, Class 1 operation performs best in transactional-intensive workloads like OLTP, web, or data warehousing.
- **Class 2** A switched bandwidth sharing connection (considered connectionless) providing a form of time-slicing algorithms that share bandwidth, thus enabling multiple exchange activities to take place. Although Class 2 service does have acknowledgment of frame delivery, frames do not have ordered, guaranteed

delivery, and thus must wait their turn in the linking schedule to perform the reordering of frames within the routing mechanism set up by the vendor. Class 2 (illustrated in Figure 14-10) demonstrates its value in typical application I/O workload situations. It can handle the transaction rates an OLTP or time-dependent system might throw its way.

- **Class 3** A connectionless buffer-to-buffer flow mechanism that offers a quick way to broadcast messages within a multicast group. However, cutting out the recipient acknowledgment and buffer coherence means introducing a high risk of frame loss. Class 3 works well for specialized applications that broadcast requests to devices. Without the retransmission of the request in other class processing, this limits the useful value to simple data replication and browsing/searching acknowledgment.

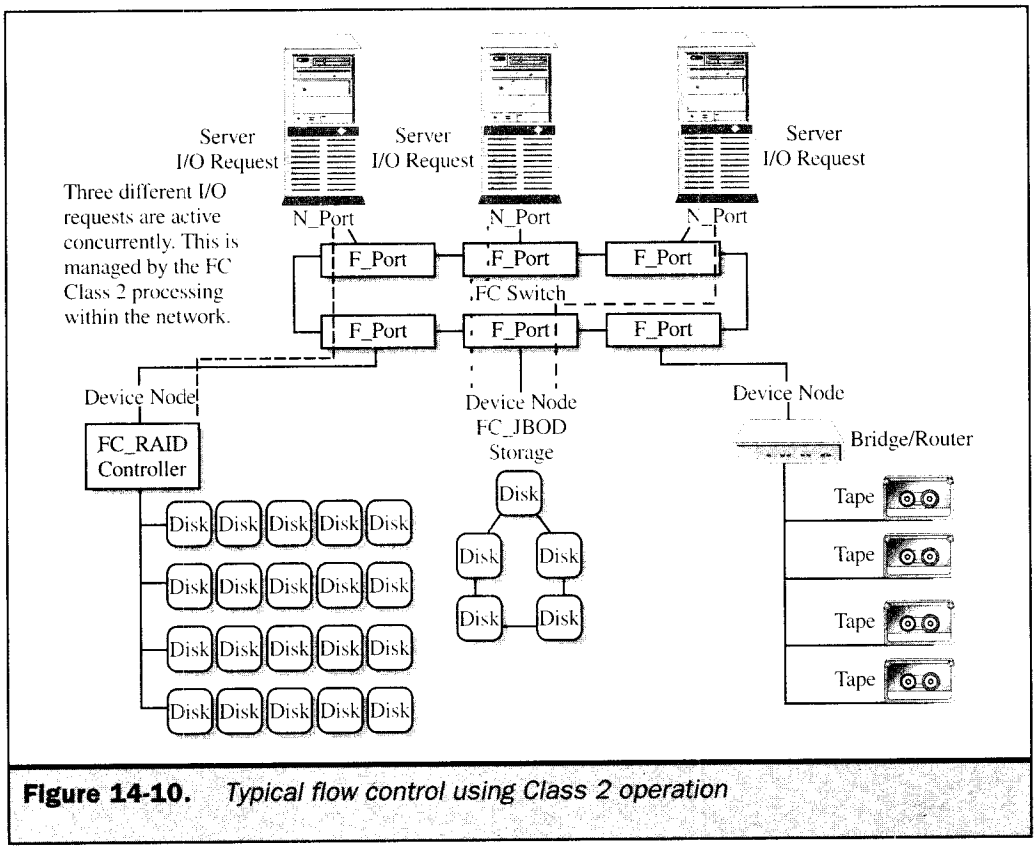


Figure 14-10. *Typical flow control using Class 2 operation*

SAN Hardware Considerations

As you might have gathered by this point, SAN hardware is made up of multiple components with many variables, specifications, and options unique to each. Most hardware decisions are made based on the features and standards supported by a particular component's vendor. It is important when heading into a purchasing decision that you eliminate as many unknowns as possible.

Of the billions of things to keep in mind, first and foremost will always be ensuring that total I/O workload matches SAN resources. (Chapter 17 lays the foundation for I/O workload planning, while Chapter 18 sets out some guidelines in estimating basic SAN resources.) Consider the following.

Match server hardware- and software-supported releases with proposed storage and SAN configuration components. This will ensure the correct operating system level, HBA, and switch fabric release levels for compatibility. Also, verify the compatibility and features of the switch against the type of I/O workload you are supporting. Ensuring you can configure Class 2 support, at a minimum, will be important as you implement the workload processing into the environment.

It is paramount that a planned SAN configuration be compatible with the operations you are planning. If you are going to use an external tape library, ensure sufficient port and network resources for transmitting data across the SAN and LAN network. If instead you are going to employ an integrated tape library, understand the compatibility issues surrounding the additional equipment. Whether integrated or discrete, routing hardware must come under the same scrutiny as every other component in the configuration.