Chapter 18

Applying the SAN Solution

The SAN architecture, as previously discussed in Chapter 17, provides a level of storage scalability and capacity beyond traditional I/O infrastructures. SANs provide a rich set of opportunities to apply to several workloads that are standard within today's data center. Among these workloads are OLTP, web services, and data warehousing. OLTP, a foundation of most businesses by supporting their daily operations, is a significant consumer of storage capacity with strict access requirements. The increasing use of web services to provide an array of Internet transactional messages between customers, retailers, and suppliers, enables a significant value benefit to all involved; however, it pushes the performance and scalability envelope of the traditional I/O infrastructure. Data warehouse applications and their related brethren, the data mart, always move the storage capacity bar to its limits as the movement of data within these applications accelerates the need for wider access bandwidth.

This chapter will discuss the guidelines for estimating SAN configurations and explore key concepts concerning the deployment of SANs into these standard workloads. Key among these will be the identification of port requirements necessary to sustain a particular workload, as well as particular issues associated with the often-disparate processing characteristics of OLTP, web services, and data warehousing applications.

SAN Workload Characterization

When characterizing workloads for a SAN, it's helpful to consider the inherent value that Storage Area Networks bring to the data center. This cross-application infrastructure enhancement may be the overriding factor in the justification of an initial SAN configuration that on the surface could be handled by direct connect architecture. This further demonstrates the importance of defining I/O workloads, as discussed in Chapter 17.

When describing SAN I/O workloads, it is also important to be aware of the possible integration of system-level capacity planning as well as network capacity planning for the obvious reason that the SAN architecture represents a combination of computer system and network characteristics that must work together to sustain the workloads. This requires that workloads be evaluated with both I/O processing and network configuration metrics. Even though SANs are young by contrast to other technologies, standard configurations have emerged that can be applied to most common workloads. The major categories, excluding a single switched environment, are cascading, meshed, and core/edge configurations.

These categories provide starting points for data center–specific customization and specialization. SAN expansion and derivations will likely come from one of these three configurations. A brief overview follows.

- **Cascading SAN Configuration** This configuration provides a switch-to-switch connection that allows the number of server and server devices to scale quickly. Figure 18-1 shows a simple cascading configuration with three servers and multiple storage devices using three FC switches.

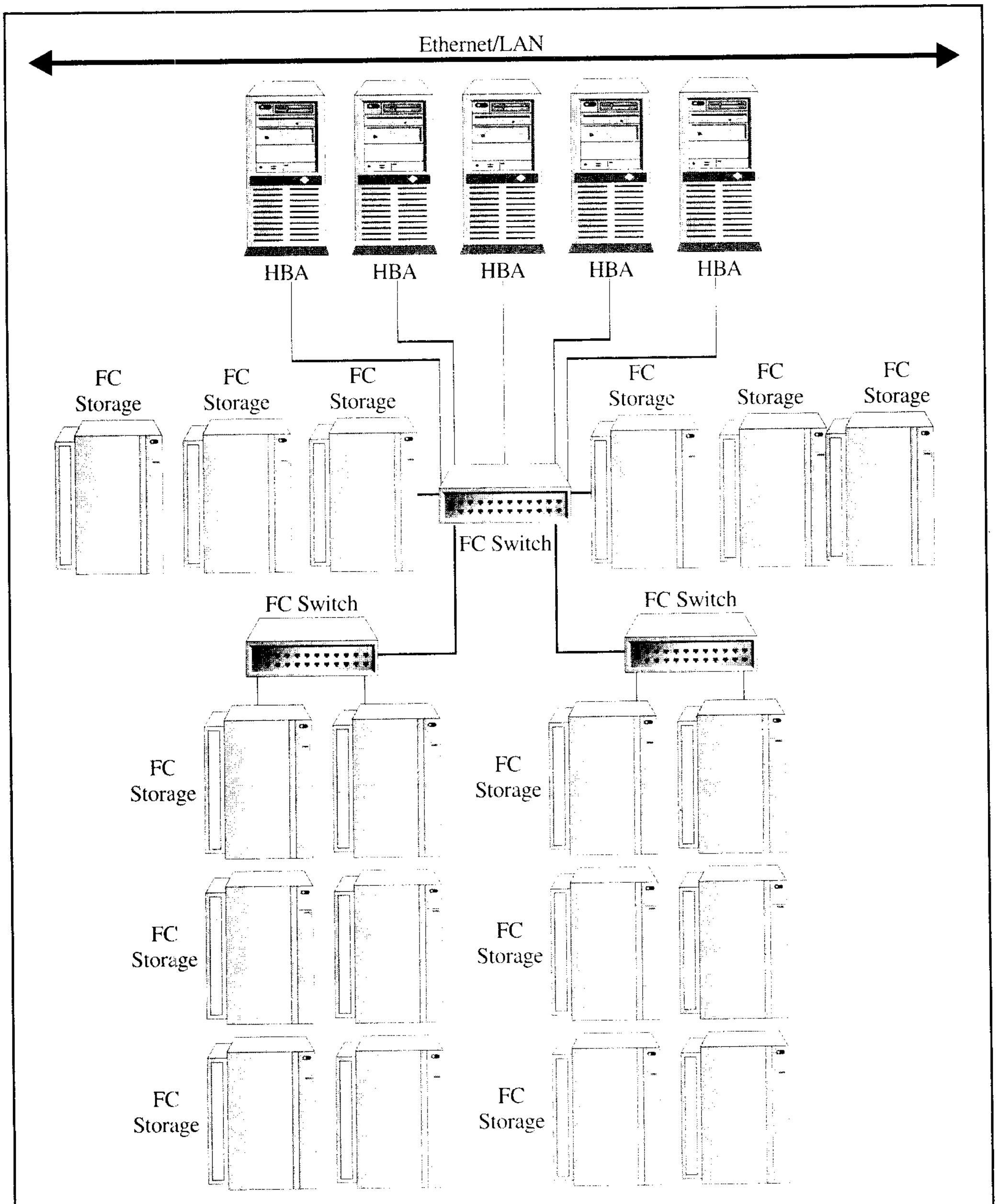

Figure 18-1. *Cascading SAN configuration*

- **Meshed SAN Configuration** This configuration provides a performance-oriented system that allows for the quickest path from server to data. Figure 18-2 illustrates how an I/O's path is reduced as it can reach the storage array without traversing multiple switches. However, it also provides the multiple connections to connect to the storage array using alternate paths in the event of heavy traffic or switch disruption.

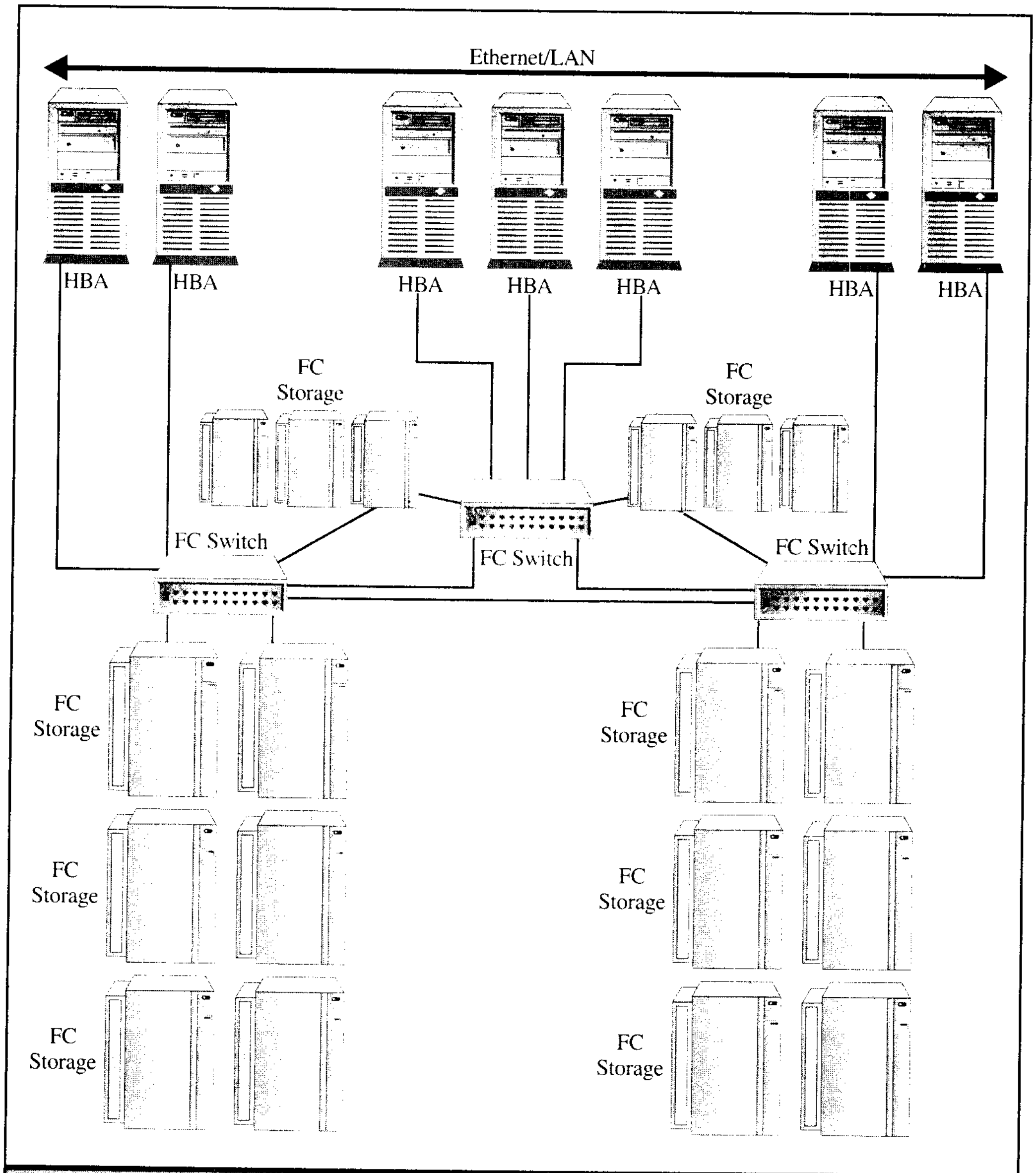

Figure 18-2. *Meshed SAN configuration*

- **Core/Edge SAN Configuration** This configuration takes into account I/O optimization, redundancy, and recovery, as shown in Figure 18-3. By far the most performance oriented, it is also the most complex in implementation and configuration.

These are all important considerations as you begin to evaluate your SAN design and implementation. Some words of caution, however, before the details of configuration complexities overcome your planning and design:

- Identify and describe the I/O workloads of the applications you expect the SAN to support.
- Understand the strengths and weaknesses of each of the major networking configurations.

Reviewing Chapter 17 will help get you started on I/O workload analysis.

Assuming you have completed a general workload analysis and have a total I/O workload transfer rate (see Chapter 17 for guidelines on estimating I/O workloads), we can estimate the number of ports required to support the I/O workloads. The sidebar "Guidelines for Estimating the Number of Ports Required to Support I/O

Guidelines for Estimating the Number of Ports Required to Support I/O Workloads

The assumptions to consider in viewing these guidelines include the following:

- The I/O workload transfer rate is available and accurate.
- The base configuration consists of FC switch ports using switched fabric.
- The base ratio available from the server to a number of data paths (for example, the number of HBAs for each server connected to the SAN configuration).

Guidelines for estimating SAN port requirements include:

- Total I/O Workload Transfer Rate (I/OWTR) / Maximum Port Transfer Capacity (MPTC) = Number of Port Data Paths (PDPs) required
- PDPs × Redundancy/Recovery Factor (RRF) = Number of ports for Redundancy and Recovery (RRPs)
- PDPs + RRPs = Total Data Path Ports required
- Total PDPs (with RRF) + Server Ports (SPs) (number of servers × number of HBAs) = Total Switch Ports required Where RRF = 30–40%; SPs = 1–4

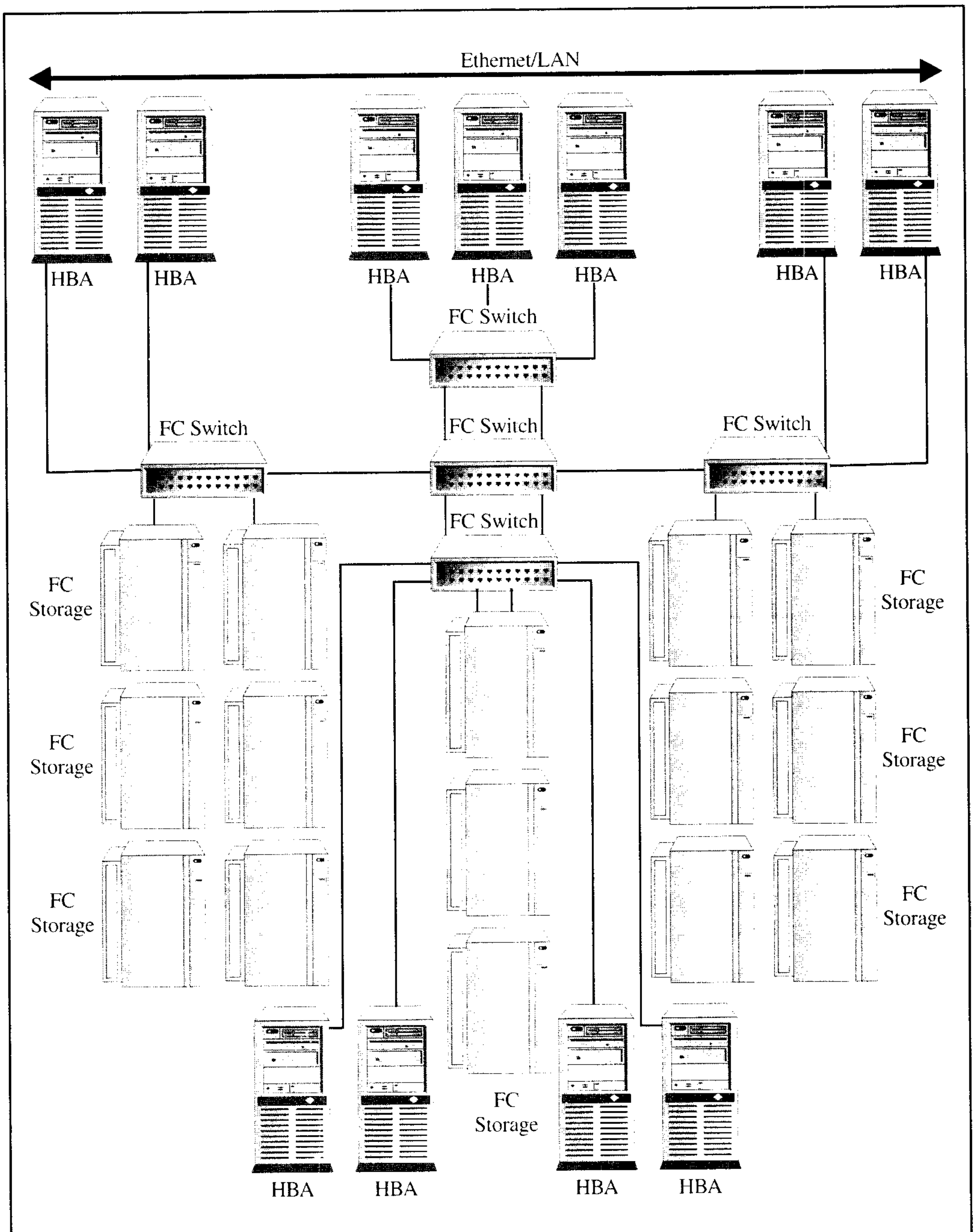

Figure 18-3. *A core/edge SAN configuration*

Workloads" describes a set of activities helpful in estimating the number of ports required.

Once the total estimated numbers of switch ports are calculated, the workload characteristics can be applied to determine the type of access and performance factors necessary. As shown in our examples, this can be an OLTP-type application, such as our banking example in Chapter 17, or a typical web transactional workload that may be used for processing warranty and service information. Finally, the ubiquitous datacentric application, the data warehouse, demonstrates unique processing characteristics and is enhanced by the basic architectural value of SAN architectures.

Switch port estimates are applied to each of these workloads, which drives the configuration into connectivity schemes that best suit the workload. As discussed previously, we then configure and distribute the port count into a cascading, meshed, or core/edge configuration. Through examples, we apply an I/O workload for each configuration and discuss the rationalization for our decision, first using OLTP to demonstrate a core/edge solution. We then move to web transactional applications supported by a meshed configuration, and finally a data warehouse using a cascading architecture.

Applying the SAN to OLTP Workloads

What are OLTP workloads? Multiple users performing a set of consistent transactions characterize online transaction processing workloads. The transaction mix will be simple to complex, but consistent in its processing requirements, usually applying an 80/20 rule (that is, 80% simple, 20% complex). The transaction type identifies its nature in terms of resource utilization, I/O content, and I/O utilization. If we use our banking application as an example, the tellers perform most of their transactions using the deposit transaction. This transaction verifies a customer account and adds the deposit to a suspense balance. The suspense files would be used as input for later off-shift processing during a batch cycle to update permanent customer records and accounts.

In this case, the simple transaction's I/O content is approximately 80 bytes of customer and account information. This is accomplished with a minimum of simple math additions and balance calculations within the application. Although small, these are executed on a transaction-by-transaction basis and require an existing file or database table be updated. Given that profile, each transaction generates an I/O operation to complete the transaction process. Meanwhile, every associated I/O completes a read/write operation to a specific disk within the supporting storage array. In the case of RAID implementation, the physical I/Os will increase based upon the type of RAID level in operation within the storage array.

Consequently, the I/O workload is characterized by many small transactions where each will require an I/O operation—multiple I/Os as RAID is introduced. As the transaction rate builds during peak utilization, the workload is further characterized by additional simple transactions adding multiple I/Os, all requiring completion within a three- to five-second time period (essentially the service level for the deposit transaction). The workload and service level demands a configuration that can handle the peak load. This requires a large number of data paths to handle the I/O transaction rate.

SANs are excellent choices for OLTP workloads because they can provide more data paths than any other storage model. Given the flexibility of the storage network, the scalability of the I/O workload can be enhanced without the addition of server resources.

SANs also provide the additional benefit of data partitioning flexibility. Given that most OLTP applications leverage the services of an RDBMS, the ability to utilize the database's partitioning schemes intra-array and inter-array provides the additional balance necessary to physically locate data for OLTP access.

Most OLTP applications support an operational aspect of a business. Therefore, data availability is important given that the day-to-day activities of the business depend on processing transactions accurately. Any downtime has an immediate effect on the business and therefore requires additional forms of redundancy and recovery to compensate for device failures and data integrity problems. SANs offer an environment that enhances a configuration's ability to provide these mechanisms in the most efficient manner.

Additional discussion regarding recovery and fault-tolerant planning can be found in Part VI of this book. However, it's important to note that each of the workloads discussed will have various degrees of business impact and therefore require different levels of recovery management.

The Data Organizational Model

The use of relational database technology (RDBMSs) defines the major I/O attributes for commercial applications. This provides OLTP workloads with a more defined set of processing metrics that enhances your ability to estimate I/O behavior and utilization. The use of the relational database has become so accepted and widespread that its macro behavior is very predictable. In recognition of this, additional consideration should be given to I/O workload characteristics relative to I/O processing requirements such as caching, temporary workspace, and partitioning.

Don't make the mistake of overlaying the storage infrastructure too quickly. The consideration of recovery scenarios and availability requirements should be considered at the macro-level first, followed by decisions on how to handle workload specifics. Particular subsets of a workload, such as a complex set of deposit transactions, may require different resources or configurations.

OLTP workloads using an RDBMS can be supported through RAID arrays using level 5 configurations. This provides both redundancy and fault tolerance needs. RAID 5 allows the storage array to continue to function if a disk is lost, and although running in a degraded mode it permits dynamic repair to the array, or, if needed, an orderly shutdown of the database itself. Additional storage system features, such as caching strategies and recovery models, require due consideration given the unique nature of the data contained within the relational database model.

User Access

User traffic plays a significant role in defining the number of data paths necessary for the OLTP transactions to be processed within the service level. Obviously, we need to know the estimated traffic to understand the type, number, and behavior of the data paths prior to assigning the configuration ports. We basically need three types of information from the end users or their representatives—the application systems analysts. This information consists of the number of transactions, the time period within which the transactions will be executed, and the expected service level.

The deposit transaction used in our example banking application comes in three forms: simple, authorized, and complex. Simple requires an update to a customer table, while authorized requires an authorized write to the permanent system of record for the account, necessary for a double write to take place when accessing more than one database table. Complex requires the authorized transactions but adds an additional calculation on the fly to deposit a portion into an equity account. Each of these have a different set of data access characteristics even though they belong to the same OLTP workload.

Service levels become critical in configuring the SAN with respect to user access. This places our eventual, albeit simple, calculations into a framework to define the resources needed to sustain the amount of operations for the OLTP workload. From our initial information, we find there are two goals for the I/O system. First, the banking applications data needs to be available from 9 A.M. to 5 P.M. each workday. Second, the transactions should complete within a time frame of 5 seconds. It is important to note that although the I/O can and will take up a great deal of the response time factors, network latency issues should not go without consideration.

In addition, the configuration must support a high availability uptime, usually expressed in terms of the percentage of downtime for a period. Therefore, 99 percent uptime requires the configuration to only be down 1 percent of the time. For example, if the OLTP time period is 12 hours each workday, the downtime cannot exceed 7.2 minutes every day. The users expect availability of data, which is a foundation requirement for meeting the response time (in other words, the data must be available to process the transactions, as well as be available during the batch cycle to process information for updating other database tables). This also forms the requirement for supporting the reliability factors and defines the type of storage partitioning and structure suited to providing this.

Data Paths

Next, we need to understand the data highway required for this workload. This necessitates breaking down our first two categories into a logical infrastructure for the workload. By comparing the data organizational model (that is, the database type and its characteristics, as well as the byte transfer requirements) with something called the *concurrent factor*, we can begin to formulate the number of data paths necessary to meet workload service levels. The concurrent factor provides us with a minimal and logical set of paths to sustain our service level, given the probability of all tellers executing deposit transactions at the same time.

This estimate provides a more accurate picture of the amount of resources needed to sustain the service level in real time. In reality, the probability of all tellers executing a deposit transaction is actually quite high during peak hours, and could be calculated at 90 percent. Therefore, for each time period, 90 percent of the tellers would be executing a deposit transaction. Using our previous calculation, we estimate the mix of simple, authorized, and complex deposit transactions to be 80, 15, and 5 percent, respectively.

We can develop our own OLTP SAN model by using the preceding considerations as well as the guidelines for I/O workload port estimates. With our banking example, we estimate that 116 switch ports are required to support the OLTP application. From this information, we can begin to model the appropriate design topology for the I/O workloads. Our previous definitions lead us to a core/edge configuration, which supports the maximum number of data paths into the storage arrays, while minimizing the length of each transfer for each I/O (for example, the number of interswitch hops is kept to a minimum). This is also the configuration that provides the best redundancy when alternative paths are needed in the event of a port interruption or switch failure. Cost not withstanding, we can now evaluate configuration options with this logical model in terms of switch types, port densities, and recovery options.

The value of workload identification, definition, and characterization becomes evident as we move into the OLTP SAN implementations.

The Design and Configuration of OLTP-Based Workloads

Figure 18-4 shows an example of our core/edge configurations supporting the OLTP application. This configuration is comprised of four FC switches, 15 disk arrays, intersystem-link ports, and an integrated FC-SCSI bridge into a tape library. This supports our workload analysis estimate of ports using three servers with two HBAs, respectively. It assumes a relational database that is capable of partitioning among the storage arrays and leveraging a RAID 5–level protection scheme within each array.

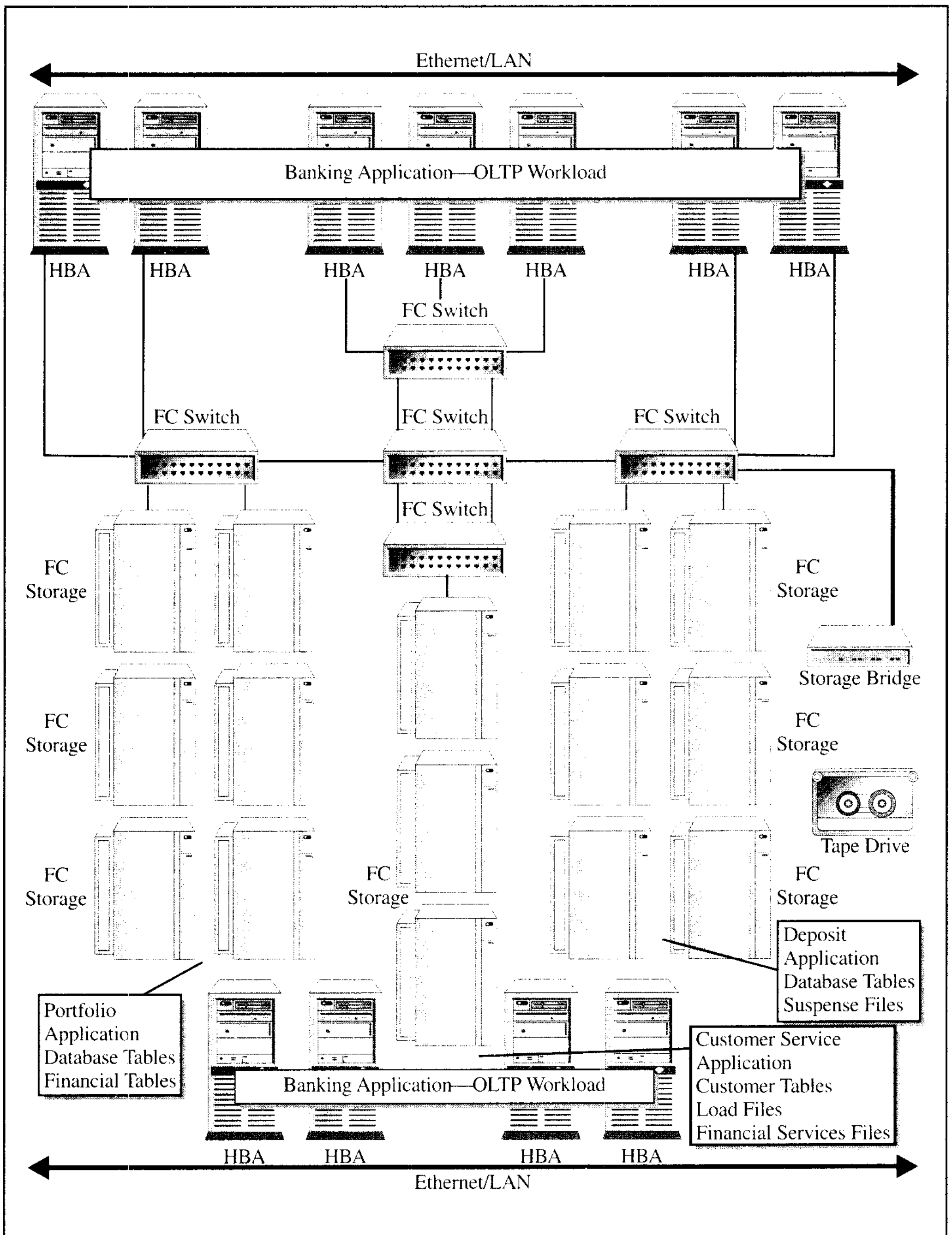

Figure 18-4. *An OLTP workload using a core/edge SAN configuration*

Applying the SAN to Web Transactional Workloads

Workloads generated from Internet or intranet sources can be very complex. Internet applications can generate transactions from simple reads of web pages to complex queries and updates of database structures. Based upon the processing architectures of web server software, which enable interactive web services, many of the workloads will be almost entirely transactional in nature. However, within this transactional model, another type of transaction type will emerge—the messaging transaction.

Though messaging transactions are processed in real time, they will have a different set of processing characteristics than typical OLTP transactions. Messaging provides an asynchronous form of processing where the client application who has submitted the messaging transaction relinquishes any synchronous connection with the corresponding application. The most familiar of these is e-mail, although many other Internet transactions are message-based. This produces a different set of processing characteristics and certainly a more complex set of I/O operations.

Note *The emergence of asynchronous processing of applications also produced an additional category of software termed middleware. The middleware, which has been used in both traditional interactive application processing architectures as well as Internet/web-based applications, provides the mechanisms for queuing a transaction for later processing. This can be for effective resource utilization or handling larger interactive transaction requests where synchronous processing is impossible. The use of middleware within applications can be a key indicator that storage networking can also be an effective supporting strategy given its architecture to handle multiple workloads and, in the case of middleware, handle multiple middleware queues.*

Messaging workloads generate asynchronous I/O operations. This is actually a very simple model where an application transaction is submitted, and a request is queued for later processing. The client, or *initiator*, of the request is notified that the transaction has been accepted for later execution. The processing is executed at a later time or when required resources become available or are provided. All this is accomplished in a manner that doesn't require the user, or initiator, to wait for the transaction to complete and can continue to execute other transactions. Although the characterization of the messaging workloads can be estimated similar to OLTP as we demonstrated previously, care must be taken to account for delayed I/O and resulting byte transfer requirements that may go unaccounted for when estimating total I/O for web transaction workloads.

Note *The difference between client and initiator is an important distinction because other software and internal storage bus processes operating on storage devices are initiators of messaging or asynchronous operations.*

Web Internet transactional workloads, which we'll refer to as WIT workloads, also contain typical synchronous transactions, much like those described in our OLTP examples. Many older WIT workloads use file systems for their data organizational models and specifically their web-based file formats. However, the RDBMS vendors have made significant advances in providing a relational model database for web servers. Consequently, many new applications utilize a relational database demonstrating characteristics similar to databases supporting OLTP processing. The fundamental I/O resource requirement details continue to be contained in the I/O activities relative to user access, data organization, and data paths. An analysis of each of these areas begins to form a picture specific to WIT workloads.

In general, WITs require flexibility and multiple paths to sustain their I/O workloads. However, they are not as sensitive to multiple hop degradation given the amount of messaging contained with the workload. Therefore, a meshed configuration can provide both transactional synchronous support through single-hop data paths while messaging data can sustain longer paths due to their asynchronous processing.

The Data Organizational Model

WIT workloads simply use the OS file system as the data organizational method. This provides a straightforward view when determining byte transfer rates for WIT I/Os given the format of web file structures. However, this must be augmented when considering the amount and type of delayed I/Os generated from messaging transactions. Similar to the transactional structure of OLTP, the typical WIT I/O may be small but the random nature of arrival rates and unpredictability regarding the number of users provides a challenging configuration problem. This defines the need for placing WIT workloads in the most flexible environments possible.

User Access

Defining user traffic is the most challenging activity in estimating WIT workloads. In web-based processing environments, the classic time periods do not apply. This is especially true in retail-oriented web sites that rely on customer interaction for sales and which are open 24/7. The requirement to provide adequate service levels in an environment where user interaction crosses worldwide time zones means estimating WIT workloads becomes problematic and challenging at best.

It is not impossible, however. Many software packages and services provide an accounting of user access and traffic analysis. Though most are oriented toward information access and purchase patterns, there is sufficient information to provide a macro estimate of I/O workloads. User access in terms of time periods, time zones, and further seasonal factors demonstrate a SAN configuration's ability to provide a flexible storage environment that supports OLTP and messaging types of I/Os.

Data Paths

The number of data paths, as with any workload, is key to the performance of WIT workloads. The influence of FC frame capacities may become a factor when analyzing switch port capacities, given that the nature of OLTP may prove to be overkill in some web transactions. However, in many cases, the FC frame capacity may become the required transport, given the increasing I/O content of web transactions which continue to grow with text, image, and other unstructured data needed for transport. Even though the SAN I/O comprises only a portion of the overall response time factor, given the move to larger Ethernet capacities in order to support client IP networks, the matching components have begun to come together.

Note *FC frame payloads transport a maximum capacity of user data (approximately 2k).*

Data paths for the WITs in a complex web environment, comprised of both OLTP, messaging, and batch loads, will likely surpass the number for typical OLTP. However, the gating factor to this continues to be the reliance of the web server file system. It's important to keep in mind that the probability of multiple data paths may have little or no effect if client requests are queued at the file system. Consequently, there may be a need to consider the following:

- The ratio of channels per server (for example, the number of HBAs per server)
- The need for SMP-enabled servers (meaning more concurrent processing)
- Ensure workload balancing can be used in some form (for instance, in a workload balance software package, processing affinity options, or process prioritization)

More detail on these options can be found in Part VI of the book.

The Design and Configuration of Web-Based Workloads

Figure 18-5 shows an example of a mesh configuration supporting the web-based workload. This configuration is comprised of three FC switches, 18 disk arrays, intersystem-link ports, and an integrated FC-SCSI bridge into a tape library. This supports our workload estimate using web-based consideration and using seven servers with two HBAs, respectively. It assumes a combination of web-based e-mail files, a transactional relational database supporting customer retail transactions, and a complement of web pages. This supports both transactional activity through the meshed configuration where numerous paths exist to the data while the messaging workload is supported by sufficient data paths with consideration for delayed I/O.

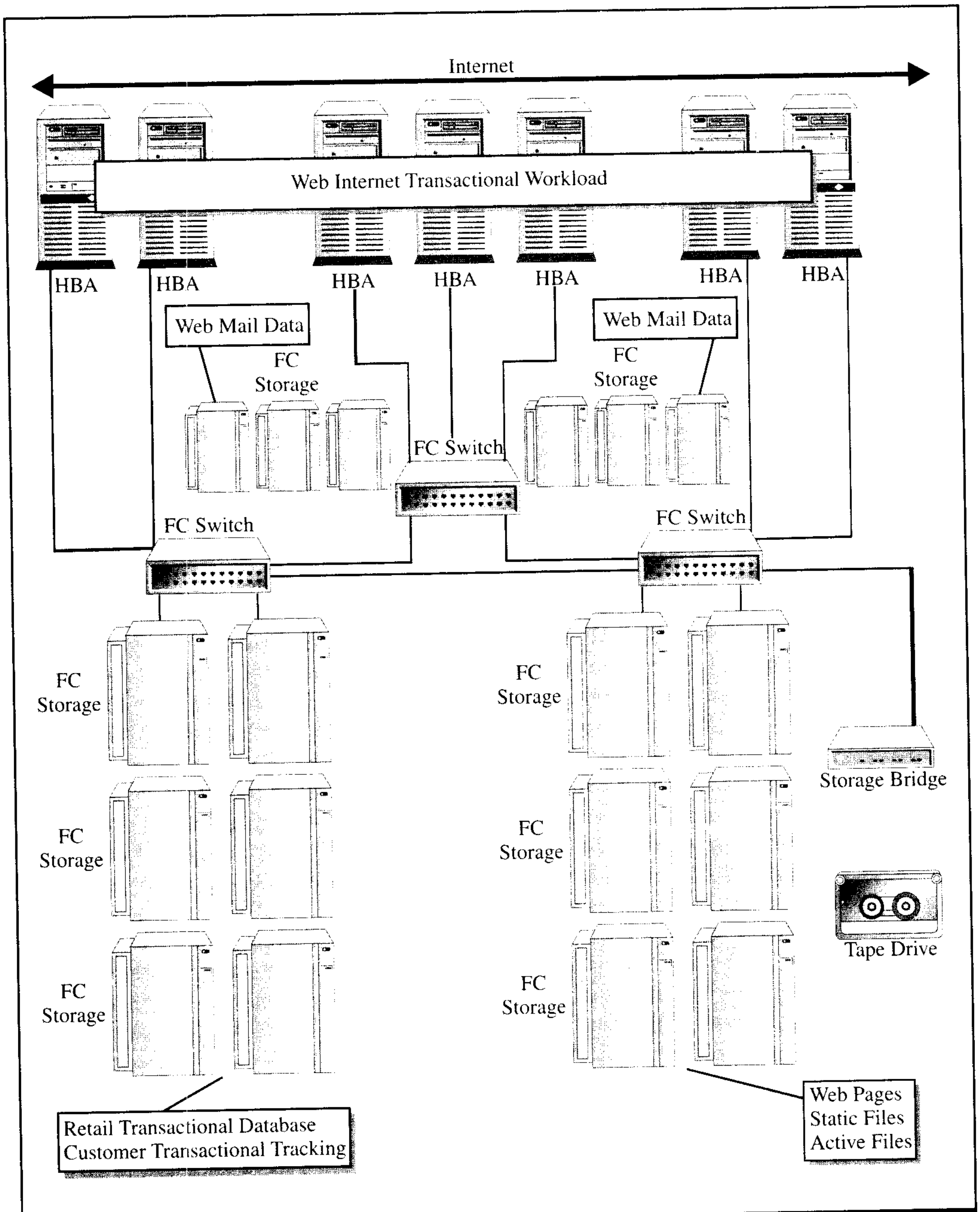

Figure 18-5. *A web Internet transactional workload using a meshed SAN configuration*

Applying the SAN to Data Warehouse Workloads

Data warehouses provide strategic and tactical business information used to direct business planning and forecasting analysis. Data warehouses are built from subsets of operational databases, typically OLTP workloads that form a historical database of various business activities. Users query these large databases in the form of "What if" transactions that range from simple to complex questions. The answer sets of these queries can be extremely large, complex, and form the basis for additional subsets of databases that warrant further analysis.

A spin off of data warehouses is the data mart. Theoretically, data warehouses are based upon corporate information and are used to drive corporate business decisions. Data marts are subsets of data warehouses that are used to drive lower-level business decisions within the divisional or departmental levels. This is important because the size and I/O content will be driven by use of the data warehouse at the corporate level or divisional or departmental level. Given that the true nature of data warehousing is tied to the functions of the relational database, there are many database rules, guidelines, and practices that determine its design, operation, and usage. However, most of these are outside the scope of this book and although important, only those that provide us with clues to their I/O workload analysis will be mentioned.

As you may have experienced or surmised, data warehouses can be very large. Given their sources are the continuing updates from daily business transactions, their storage utilization has been the most comprehensive. The capture of operational database records—known as the extraction, transformation, and loading activities—are driven by the complexities of the data analysis application. In other words, the more comprehensive the analysis application, the more data is required. Many data warehouse configurations are moving into the multiple terabyte range and, consequently, provide the foundation for why data warehouses and data marts are good candidates for SANs. Of all the workloads examined so far, the data warehouse provides the most complex I/O activities that we will discuss.

Data warehouses are built around relational databases; therefore, they are under the control of the user's database system of choice. Moreover, this defines the need to provide large storage capacities that scale well in terms of pure size and I/O content. This is due to the activity of the data warehouse transaction. The DW transaction in its simplest form accesses information from multiple database tables which span multiple records—For example, in order to analyze the number of customers with account balances over $1,000 for an entire year with credit card balances requires the database system to access a customer table, deposit accounts table, credit card accounts, and so on. Without going into the specifics of the database design, we see that even a simple query generates multiple I/Os, each with a large set of records.

DW transactions are serviced on a synchronous and asynchronous basis. Consequently, it's important to understand the mix of transactions and relative service levels. This brings up an interesting phenomenon as both online and batch-oriented operations can process simultaneously within these environments. Given that these are disparate workloads in terms of their resource requirements, it becomes a challenge to configure an I/O system capable of handling this set of complexities.

This is where SANs come in. SANs provide the most comprehensive solution for data warehousing since the MPP database machines. SANs, through their increased addressing capacity, provide an extensible and scalable solution to configuring a very large database. Moving well beyond the addressing limitations of SCSI bus configurations, the need to supply beyond 16 addressable storage devices has long been a requirement in DW environments. SANs support a comprehensive workload with considerable I/O content given the increased FC frame payload capacity for each I/O. Finally, the flexibility to isolate disparate I/Os through SAN operations and configurations can sustain a disparate and complex workload.

Consequently, taking into account the mix of disparate workload characteristics, there needs to be a balance between transactional activity, update activity, and data currency. These should be identified in the workload analysis. The subsequent resource requirement details can be estimated in the major areas of I/O activities (for example, the data organization method, data paths, and user access).

The Data Organizational Model

The use of a relational database system as the data organizational model for the data warehouse is guaranteed. This not only defines a set of metrics that allows for greater detail in determining the byte transfer rates, but also the physical placement of data within the storage devices. Working closely with a database administrator greatly enhances the SAN design and operational factors. As discussed earlier, there are three key database metrics to consider. These are database size, multiple table accesses, and updates.

Database size determines the number of storage devices necessary for pure capacity. Although important, physical placement becomes a critical factor. Partitioning within the database, meanwhile, is key to the type and transaction mix. Consequently, databases that span storage arrays have needed consideration in determining port capacities and access at the switch level.

Depending on the DW transactional mix, the ability to provide adequate performance for accessing I/O concurrently across the database is required. Databases partitioned within storage arrays (which is a given), and those that span storage arrays (a good possibility), need to be analyzed in order to provide sufficient data paths that meet required service levels. Just as important is the ability to isolate the data paths in terms of concurrent access to batch operations that may be taking place simultaneously.

The significant challenge is the configuration attributes that handle the updates. DW sources are operational databases that handle a company's day-to-day operation. Copies of subsets of data from these databases are used as updates to the DW databases. The operational databases may or may not be part of the SAN configuration. In most cases, they aren't; therefore, care should be taken in how the update process is executed. Usually, this is a batch process performed during an off shift. If this is the case, an analysis of the options for getting the data into the DW is key to the success of SAN operations.

User Access

The one break you can expect from DW configurations is that the user community supported is generally limited. Although somewhat obvious, since there are only a limited amount of company personnel assigned to strategic planning, the user base is likely to grow significantly as the success of DW applications demonstrate their value. Although user traffic may be limited, keep in mind that the I/O content of answer sets can be very large. This is one area where applications like DW push the boundaries of FC transfer rates.

If we consider our previous example of analyzing the number of customers with accounts of $1,000 for the year, with credit cards, and who have lived in the same place for more than three years, we may find that the answer set exceeds a gigabyte of data. Dealing with such a large transaction, the number of I/Os generated requires sufficient resources to meet the expected service level and that may be a real-time requirement. However, if only two users generate this requirement per day, this will begin to put the requirement into perspective in terms of analyzing the total byte transfer rates.

Data Paths

Processing transactions that have gigabyte requirements requires multiple data paths. In addition, the necessity to physically partition the database across storage arrays requires more data paths. If we augment the requirement further with database updates that may need to be performed in isolation, the number of data paths becomes significant.

Although the number of data paths needed is dependent on the service level, the ability to access multiple terabytes of data using complex queries in a reasonable time period will drive the number of data paths required by a SAN. The alternative is a highly complex MPP environment using sophisticated parallel processing functionality.

The Design and Configuration of Data Warehouse Workloads

Figure 18-6 shows an example of our cascading configuration supporting the data warehouse application. This configuration is comprised of three FC switches, 18 disk arrays, interswitch-link ports, and an integrated FC-SCSI bridge into a tape library. This supports our workload estimate using data warehouse metrics and considerations, and supports the five servers attached to the end-user LAN. It assumes three relational database systems supporting two warehouses, the Financial Analysis System and the Financial Planning System. Note that the Financial Analysis data is large enough that it is partitioned into North American activity and European activity. The outstanding issue in this configuration is the update process that occurs with operational databases that are not part of this SAN. Considering our initial example of the banking application from Figure 18-3, an interesting problem to contemplate is the operational activities necessary in updating the data warehouse from SAN to SAN. This will be discussed in Chapter 20, which concerns integration issues.

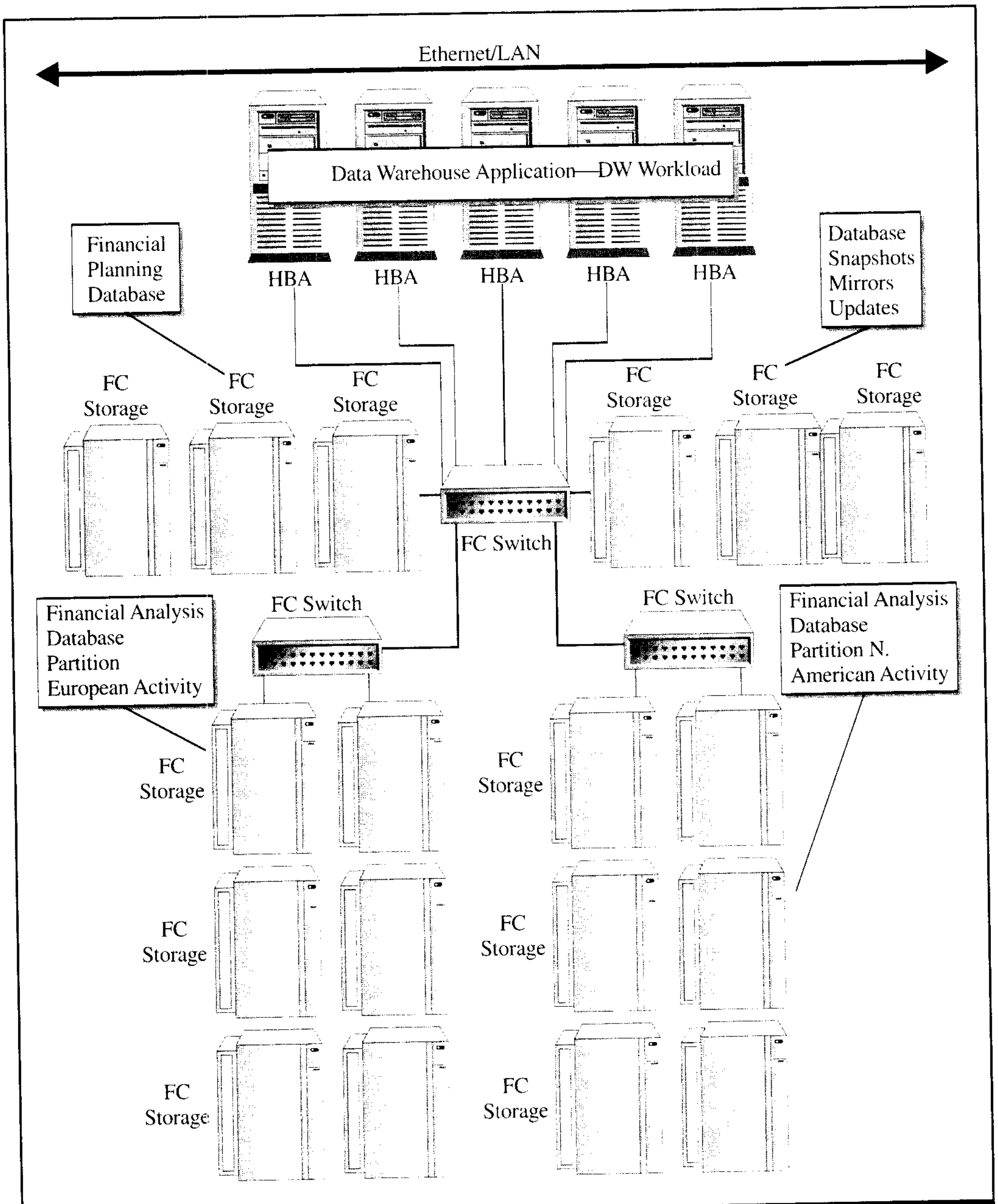

Figure 18-6. *A data warehouse workload using a cascading SAN configuration*

Conclusions on Applying SAN Solutions

Three major points have emerged from our discussion of applying workloads to SANs. First is the SAN's scope of configurations needed to handle the most common set of workloads within the data center. Second is the flexibility of the configuration which enables both transactional, messaging, and batch types of workloads. In some cases, as in data warehouse datacentric loads, two diverse workloads can be processed concurrently while providing a manageable configuration necessary to meet service levels.

Points to Consider Before Proceeding

It should be noted that SANs consume both manpower and dollar expenses, which in many cases are over and above your existing direct-attached configurations. Cost notwithstanding, the following additional considerations should be analyzed when contemplating the use of SANs within a data center.

- **Enterprise Workloads** SANs are justified using enterprise-level workloads that have resource requirements which exceed the scope and functionality of direct-attached solutions, or which exceed the capacity of a NAS solution.
- **Integrating Systems and Networking Skills** SANs require existing personnel be educated about them—there's no getting around it. However, storage expertise, coupled with network expertise, will best facilitate capacity planning, design, and installation activities.
- **Plan for Open-Ended Solutions** SANs allow the data center to design and plan for the long term, while making purchases for the short term. This leverages the SANs scalability and long-term viability as a storage solution.

Preparing for the Solution

All the solutions in these examples follow a macro plan. The following steps are recommended when implementing a SAN into a production environment. The macro plan is further defined by separate micro plans that describe actual task-level activities, IT assignments, and target dates. Additional steps may be required given the level of change control managed within the data center. (Refer to Chapter 23 for more details on change control processes and practices.)

Design and Configuration

Our examples show SANs supporting three common types of workloads: OLTP, web Internet-based, and data warehouse. It is beyond the scope of this chapter to illustrate the enterprise application, which will be defined by your workload planning. Typically, configurations are comprised of combinations of 8-, 16-, and 32-port FC switches, with disk arrays commensurate with storage capacities; however, it's not unlikely to surpass 20 distinct systems. Another important point we have focused on in configuration management is the inclusion of interswitch-link ports (ISLs), as well as an integrated FC-SCSI bridge into a tape library.

Test Installation

Define a test installation environment. Putting a small configuration in place provides essential first-case experiences useful in the configuration and operation of a SAN environment. This also provides a test bed for assessing future software and hardware upgrades while enabling an application testing facility.

Use the test installation to initiate a pseudo-management practice. Management becomes the most challenging activity when operating the SAN. It's also the fastest moving and most quickly evolving practice given the rapid change in software tools and accepted practices today. Detailed discussion of SAN management can be found in Part VI of this book.

Production Installation

Develop a production turnover activity where a formal change window is established. In many cases, this may need to be integrated into existing change management activity within the data center. Key among these is tracking the changes made to components of the SAN. It is particularly troublesome if you formalize changes to the switch configurations and don't upgrade critical components like HBAs, routers, and attached storage devices.

An important aspect of the production installation is the discipline surrounding establishment of a back-out practice. Because the SAN is an infrastructure in and of itself, its reliability can be problematic in the beginning, as with any new technology. However, being able to back out quickly and return the production environment to a previously existing state will save valuable time as you move into a SAN environment.

Production and Maintenance

If you have established a formal set of production turnover and change window practices, maintaining the SAN components should become manageable. The key area in providing maintenance to the SAN components is recognizing their complexities as interoperable components. Upgrading the fabric OS in a switch configuration may effect interactions with the server HBAs, which in turn may impact storage bridge/routers and other attached node devices.

Further establishing a maintenance matrix of SAN components is your best defense against maintenance ricochet, where upgrading or changing one component affects the operation of others. However, SANs are no more complex than networks, and as we've discussed several times in this book, they share many of the processing characteristics of a network. The differences you encounter will be in the devices attached to the network (for instance, the switches, servers, routers, tapes, and so on). This means that the mean time to defect recognition is longer than in traditional networks, given the fact that there are no clients directly attached that will provide instant feedback if the network is down or not operating properly.

Consequently, there is a need to monitor the operation of the SAN in as active a fashion as possible. Although we will cover this in more detail in the management part of the book (Part VI), it is important that the information gathered during this activity play an important role in problem identification, tracking, and resolution.