The Complete Reference

Storage Networks

Part II

Storage Fundamentals

Chapter 5

Storage Architectures

In Part II, we will review the basic elements and devices that make up computer storage. Although the level of information will be summary in detail, these are the components you are likely to encounter in dealing with changes to storage infrastructures brought on by storage networking. In many instances, it will be these devices, their performance, and logical configurations that are overlooked as they extend beyond the bounds of typical storage systems. As such, they are critical factors when addressing issues affecting storage performance. More importantly are the effects these elements, devices, and components will have when integrating storage networking into your infrastructure.

In Part I, the decoupling of storage devices caused us to reevaluate the necessary functions to develop an optimum data highway to the server. This quickly becomes a complex task as some storage devices become remotely connected, such as disks and disk controllers. With the introduction of networking and switching technologies, their capabilities and functionality begins to change and their operation with other components that transport data to and from the CPU can become compromised. However, understanding how data travels in traditional direct connect configurations can be complex in and of itself. It is imperative that these situations are understood both in context with data center practices and storage device principles of operations, prior to an in-depth discussion of storage networking.

Storage Elements

Storage can be defined as the components and locations where data is staged before and after it is accessed by the computer's central processing unit (CPU). The components that make up specific storage devices in today's computers are multifold, however they still have one main objective. They move data in the most efficient manner in and out of the CPU to process application program instructions.

The storage locations are multifunctional in usage and multidimensional in capacity. Figure 5-1 depicts the locations where data is stored for action by the CPU. Although you wouldn't expect to classify some of these locations as storage, their purpose is to provide a location for data, albeit temporary in many cases, before or after operation by the CPU. A parochial view of hard disk or archival storage does not take into account all the components involved in transporting data to and from processing objectives. By gaining a complete understanding of all storage locations, we systemically address all the elements that affect the supply of data to the CPU and therefore the system's potential, improving application program performance and storage device optimization.

Storage locations are characterized by their volatility and temporary nature. This is demonstrated through their effect on the data stored within each storage component. Data stored on hard magnetic disk, tape, or optical is considered non-volatile and therefore stores data permanently in context with its logical form—for example, a customer file, binary program, or an image. In contrast to these devices, there are the components such as system level cache, controller cache, memory buffers, and disk read/write

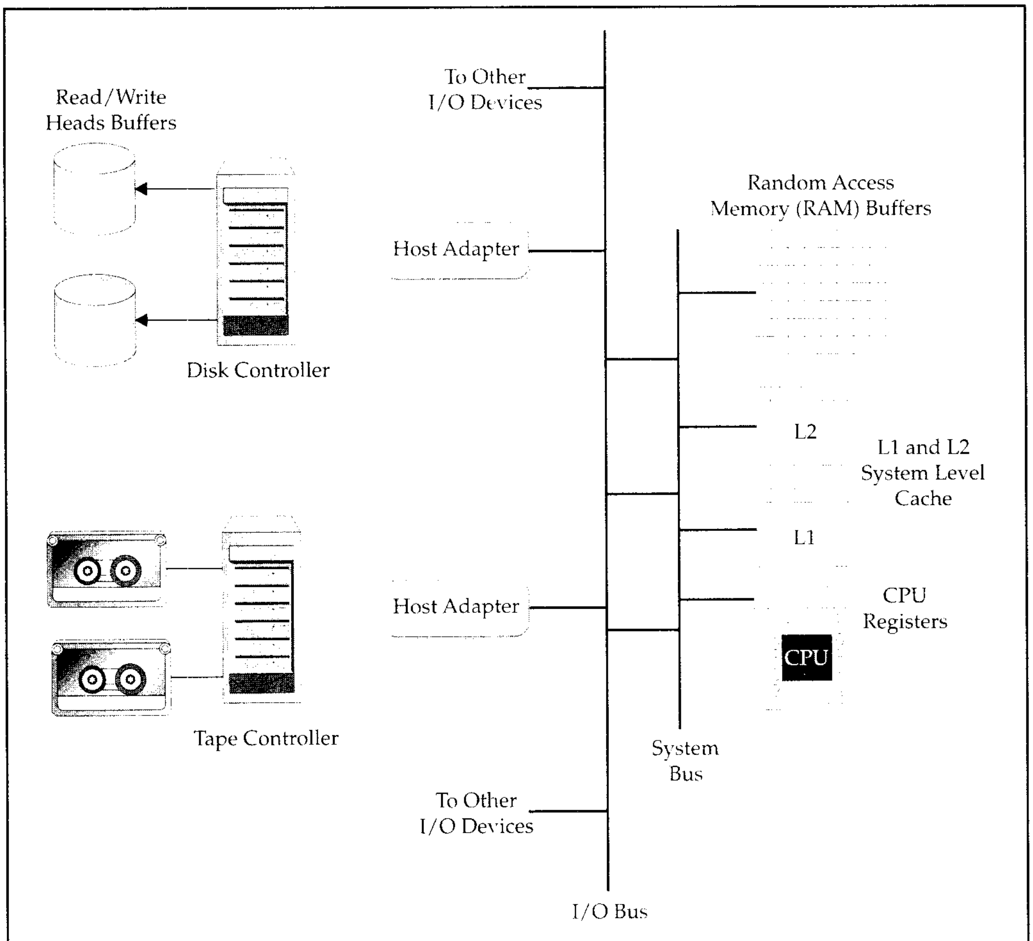

Figure 5-1. *The diversity of storage locations*

cache-buffers that are highly volatile. These components are used for only temporary locations and are staged in physical locations in increasing proximity to the CPU. The nearest component to the CPU is the fastest, and thus it is more volatile with the data it transports.

Component speed and CPU proximity drive the balance between slower and faster elements that make up this store and forward like architecture. As we saw in Figure 5-1, many of the faster components are necessary to account for the slower performance of others. Moving data into position to be accessible by the CPU requires staging areas such as system level cache so that CPU processing does not have to wait for a slow mechanical device such as a disk drive to transport a copy of its data to the CPU. The same can be said for the reverse operations; once the CPU has completed its instructions, it can use a staging area such as system level cache so it does not have to wait for the write operation to be complete out to the slower disk drive or be forced to wait the time it takes to flush and rewrite the buffer space in memory.

The corollary, or trade-off to this condition, is the economics of speed and capacity to price. Consequently, the faster the component, the higher the price for parking data. An example of the dynamics of price difference can be found in the comparison of solid-state disks (SSD) versus magnetic disks. Each can perform similar functions, however the SSD made up of logic circuits from silicon wafers in the form of a chip is much more expensive than its counterpart, the magnetic disk, which operates using mechanical devices and electronics on magnetic media. Although each stores data, the SSD device, operating electronically within a chip, is exponentially faster than the mechanical operation of the magnetic disk. Although there are many examples within the storage hierarchy, the point is the following: it becomes economically prohibitive to store large capacities on higher performing components.

Although it's possible to load small databases into the main memory of large machines, you still have the problem of keeping the permanent copy up to date in case of hardware or software error or system termination. Therefore, we find the additional corollary to speed and capacity is reliability. As system outages occur, the temporary storage locations are flushed or cleared, at which point the potential for data loss is imminent, and for most users, unacceptable. Therefore, recovery of the data that was contained within these storage locations has to be taken into account. This becomes especially challenging as we discover that some storage components are accessed remotely as networks are introduced into the connectivity scheme. As a result, many components (such as controller caches) become shared. We will examine in more detail the capability to recover various storage elements within these conditions later in Part IV, and how they will become critical design factors in Part III, regarding both NAS and SANs.

Data Storage Hierarchy

The benefits of understanding all the elements of storage is that it begins to form a foundation for working with data volatility and the effective use of the multifunctional components that data must traverse to satisfy the application. As previously stated, the effect these components have on system and application performance, such as controller cache and a disk's average access response time performance can be the difference between effective utilization strategies and overcompensation for device inefficiencies. With this knowledge, the traditional data storage hierarchy (as shown in Figure 5-2) takes on new and challenging characteristics.

The data storage hierarchy extends from the CPU to the storage media. The following highlights some of the major components that make up this data highway:

- **CPU Registers** The component of the central processing unit that stores program instructions and intermediate, prestage, and post operations data.

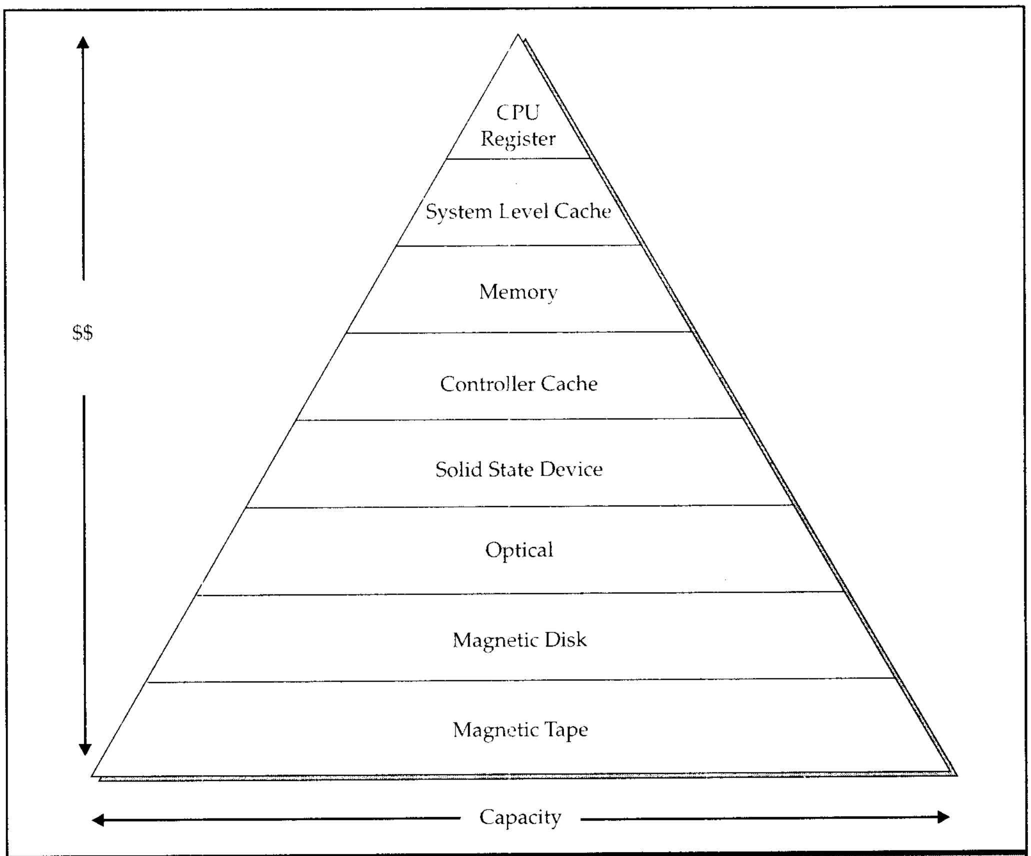

Figure 5-2. *The data storage hierarchy*

- **First Level L1 Cache** The component that provides a data staging area closest to the central processing unit; employed for prestaging and post-staging data used by the CPU instructions.
- **Second Level L2 Cache** The component that provides additional data staging areas to the CPU, but which are further away and which act as supplements to the L1 cache component.
- **Memory Buffers** Locations within the memory where data is staged for pre and post-processing by CPU instructions.
- **Direct Memory Access (DMA)** A component that allows peripherals to transfer data directly into and out of computer memory, bypassing processor activities.

- **Host Adapter Cache/Buffer** The component within a device adapter that provides staging for data that is being read or written to disk storage components.
- **Hard Disk Controller Cache/Buffer** An additional component that provides another staging area for read data that is waiting to move into the memory or L2 or L1 cache, or waiting to be written into locations within the disk storage media.
- **Hard Disk** The component that provides a permanent location for system and application data slated to be stored for access by system and applications programs.
- **Archival Storage** The locations where data is stored for later use. This may be for archiving historical data no longer needed for business or systems applications, or for copies of business or systems data needed for recovery operations in the event of system outages or other computer disasters.

Storage Systems

Storage systems have been evolving to logical stand-alone extensions to the client/server processing architecture (refer to Chapters 1 and 2). Consequently, we have storage systems that are larger, in terms of floor space, than the servers they connect to, and costlier, in terms of dollars with software and IT administration. Storage infrastructure costs make up nearly 75 percent of the cost of a client/server configuration. Why? Because we require more and more data to be available online—in other words, to be available to the end users through some form of interactive application. The result is more disks being attached to the client/server model. With the increase in the number of devices, the resulting complexity in meeting these demands is enabled through the Data Storage Hierarchy.

So, what makes up storage systems for the data center? There are multiple components that make up the storage infrastructure that has evolved into a highway for moving data. As discussed previously, the objective for these components has not changed—that is, they are used to stage data as close to the central processing units as possible, and as we have demonstrated, extend the storage highway into the servers themselves. This can drive the boundaries of the traditional views of storage into new perspectives and extend the discussion of what makes up storage systems for the data center. For our discussion, we have simplified our taxonomy to encompass the major functional configurations in which storage systems are deployed. To accomplish this, we have included in our examples the components that continued to be housed in the server. By way of definition and illustration, we have also added a new definition to our discussion, the workload. Consider Figure 5-3 when viewing our examples.

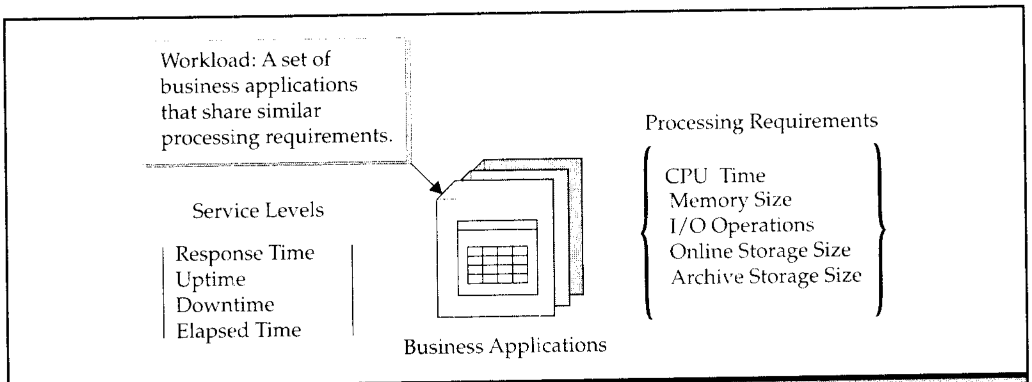

Figure 5-3. *The definition of workloads*

Typical Storage System Configurations

Online storage systems define a variety of configurations used to support applications that require immediate access to business and systems data, as shown in Figure 5-4. These configurations form the basis for support of many of the additional workloads required to operate the data center.

OLTP Workload and Typical Configurations

Online Transaction Processing (OLTP) storage systems are configured with high-performance storage hardware and software components to support business applications in processing transactions in a real-time setting. These environments employ a predefined end-user response time that has a prescribed set of business data operating within several conditional states.

Application performance is expressed as service levels within the data center. Service levels are guarantees or targets defined between end-user and data center management. These generally consist of elements of application availability (up time) and performance (response time). Additional information and discussion can be found in Part VI.

Batch Workload and Typical Configurations

Batch processing storage systems are configured to support applications that are processed in a background fashion and which don't support end-user access, but that nonetheless require immediate access to business and systems data to meet a predefined elapsed time interval for processing. These environments are often highly integrated with OLTP

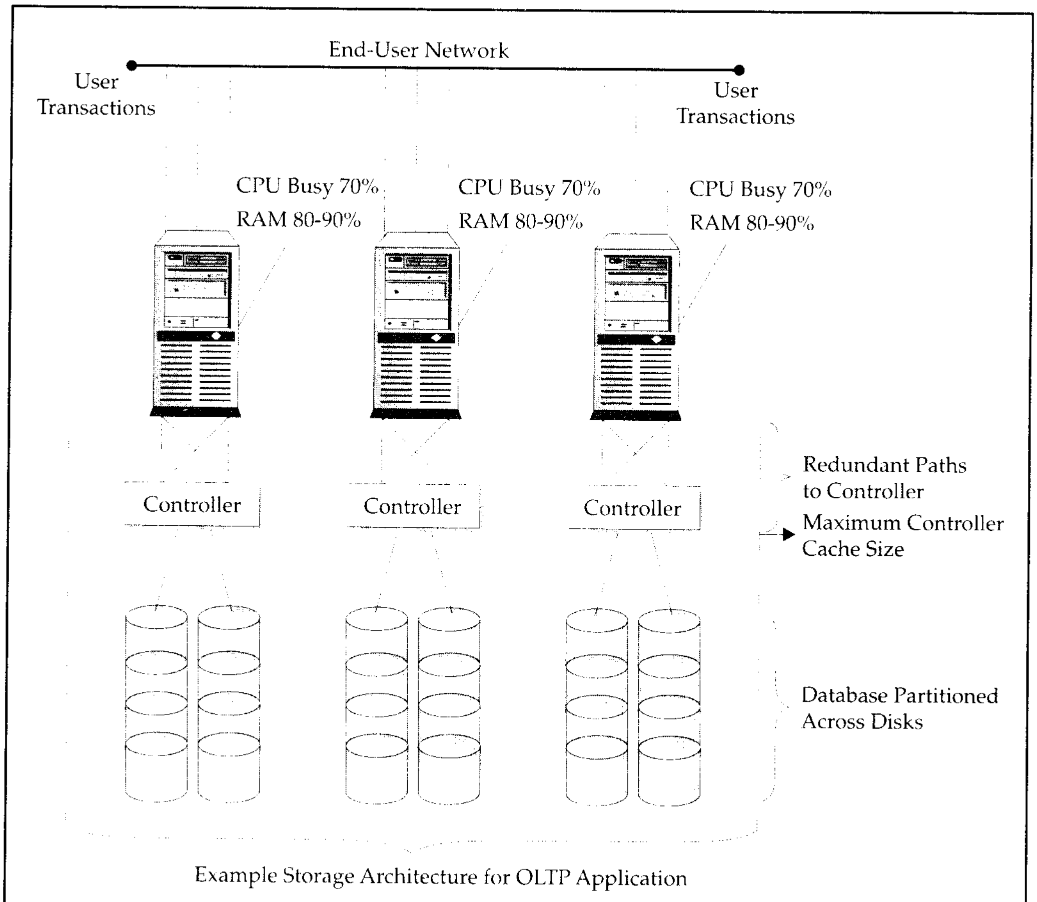

Figure 5-4. *An online storage system*

environments and process data captured as a result of the OLTP system, as shown in Figure 5-5. However, unlike OLTP systems, they are usually optimized toward high throughput without regard to transactional service levels or response times. Additional discussions in Part VI will address the interaction between batch and OLTP workloads and the impact they can have on their respective service levels.

Archival Storage

Archival storage depicts a system used to support applications that archive business and systems data no longer needed for online applications, or the copies of business and

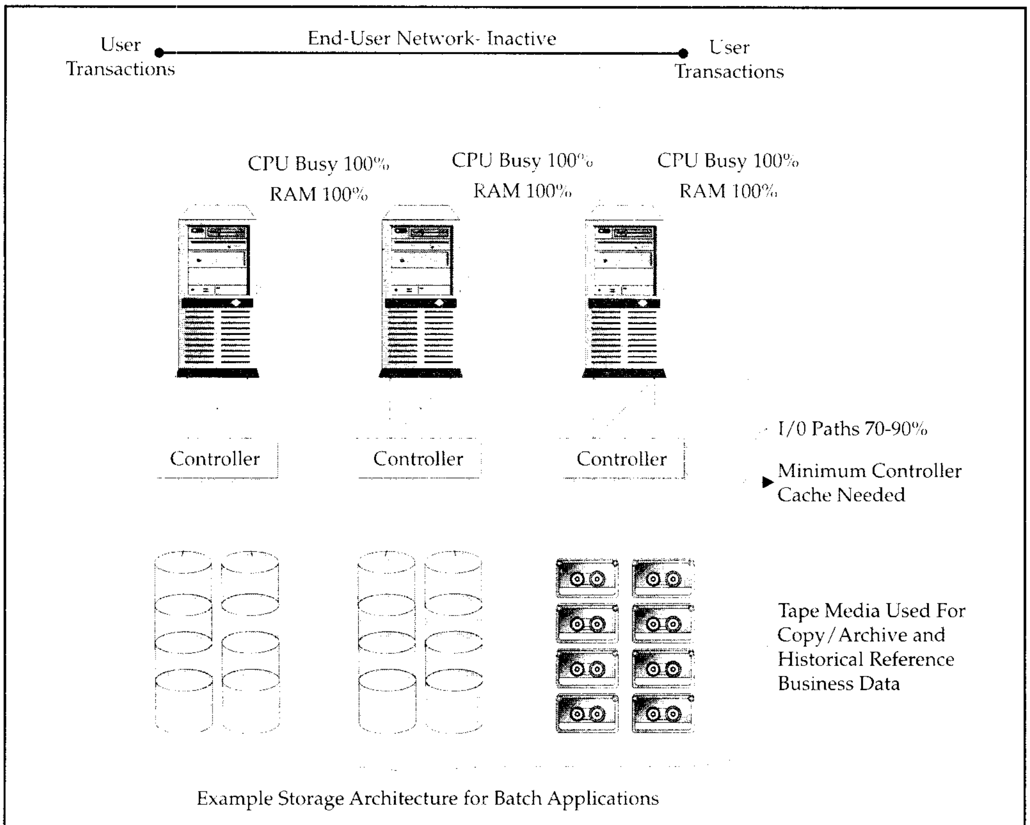
User Transactions
End-User Network- Inactive
User Transactions
CPU Busy 100%
RAM 100%
CPU Busy 100%
RAM 100%
CPU Busy 100%
RAM 100%
I/O Paths 70-90%
Controller
Controller
Controller
Minimum Controller Cache Needed
Tape Media Used For Copy/Archive and Historical Reference Business Data
Example Storage Architecture for Batch Applications

Figure 5-5. *A batch storage system*

systems data needed for recovery operations in the event of system outages, data corruption, or disaster planning.

Data Protection Configurations

Data protection storage systems are configured to support maintenance applications that copy data in both logical and segmented forms for later reconstruction. Most are processed in a background fashion and therefore don't support end-user access, but nonetheless require immediate access to business and systems data to meet predefined elapsed time intervals for processing. These environments are often highly integrated with both OLTP and batch environments and provide recovery of business and systems data in the event of various system outages. These systems are usually optimized toward high throughput without regard to transactional service levels or response times, as shown in Figure 5-6.

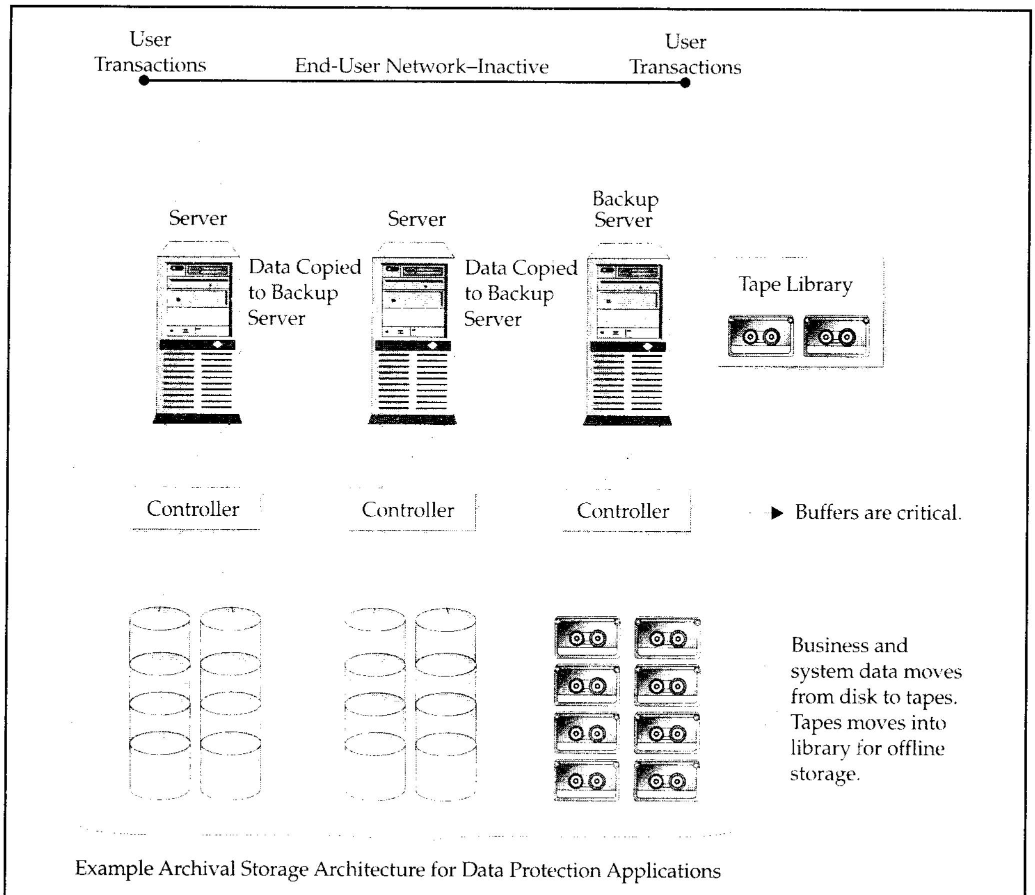
User Transactions
End-User Network–Inactive
User Transactions
Server
Server
Backup Server
Data Copied to Backup Server
Data Copied to Backup Server
Tape Library
Controller
Controller
Controller
▶ Buffers are critical.
Business and system data moves from disk to tapes. Tapes moves into library for offline storage.
Example Archival Storage Architecture for Data Protection Applications

Figure 5-6. *The data protection workload*

Near-Line Storage Configurations

Hierarchical Storage Management (HSM) storage systems (illustrated in Figure 5-7) typically support OLTP applications but are configured with lower performance, and higher capacity, storage hardware, and software components. These systems support a business application's capability to process transactions in a near-real time setting, meaning that some user transactions are asynchronous and can take longer to complete as compared to real-time OLTP response times. However, these environments run with a predefined end-user response time, that although longer, runs with a prescribed set of business data that operates within several conditional states.

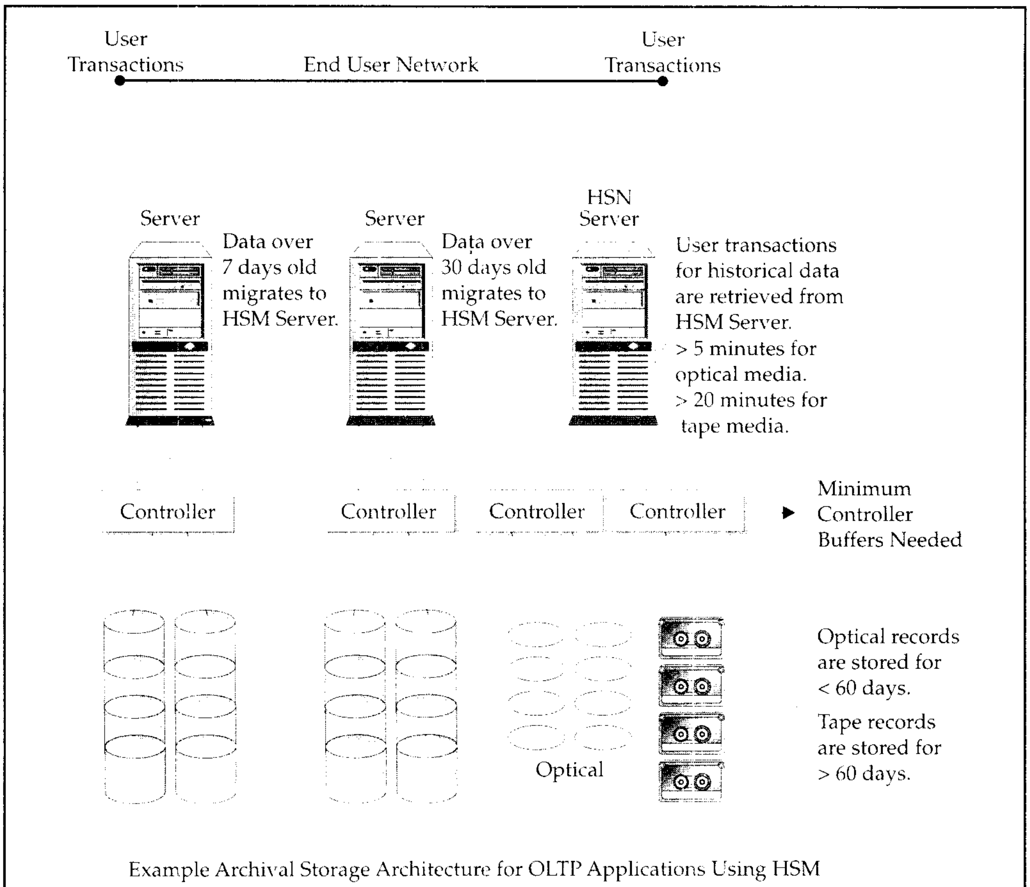
User Transactions
End User Network
User Transactions
Server
Data over 7 days old migrates to HSM Server.
Server
Data over 30 days old migrates to HSM Server.
HSN Server
User transactions for historical data are retrieved from HSM Server.
> 5 minutes for optical media.
> 20 minutes for tape media.
Controller
Controller
Controller
Controller
Minimum Controller Buffers Needed
Optical
Optical records are stored for < 60 days.
Tape records are stored for > 60 days.
Example Archival Storage Architecture for OLTP Applications Using HSM

Figure 5-7. *A typical configuration for HSM workloads*

In all fairness to the complexities of large data centers, these systems, in practice, are integrated within the existing production configurations. For example, in most cases the online storage configuration that supports the OLTP application is the same one that supports the batch cycle during an off-shift cycle. Whether this is the most effective for batch workloads is a discussion of our first corollary: speed and capacity versus price. The archive system is often integrated into systems that support some level of HSM or near-line system. Again, whether it interferes with backup jobs run at night is a discussion of our second corollary: speed and capacity versus reliability.

Consequently, when evaluating storage systems, one must take into account the multifunction requirements placed upon it. The likelihood that the most end-user sensitive configurations, such as OLTP- and HSM-like workloads, will take precedence in resources and administration is quite high. These are the systems that generally provide support for the end user's day-to-day business operations and which bring visibility to its customers.

With our global definitions and typical workloads, the online storage (OLTP) configurations will form the primary foundation for the storage infrastructure. This will be enhanced and augmented with other, albeit less time-sensitive applications, such as HSM supported applications, the application batch cycle, and last but not least, the archival of data, backup, and recovery jobs necessary for data protection, disaster planning, and legal purposes.