

Introduction

Storage networking represents a fundamental paradigm shift in how data is stored and accessed within computer systems. Consistent with any major technological change, storage networking brings with it the challenge to understand new models of storage architectures and, more importantly, the realities of integrating new solutions into the data center. *Storage Networks: The Complete Reference* was researched, designed, and developed to be a general, yet pragmatic, reference for those in the IT community who are already entrenched (or plan to be involved) in this fast moving, and at times, difficult-to-understand storage foundation for the future.

I believe storage networks to be the first major enhancement in computer architecture since the client/server revolution began in the late 1980s. Since that time, the need to store greater amounts of data has never ceased. The success of the client/server phenomenon pushed the processing power of the server component technologies to new levels that in turn exceeded the limits of storage devices and their related connectivity components. This situation drove systems and storage vendors to develop an ever-increasing, yet short-lived, set of disk and tape products that reflected the exponential appetite of end users for increasing capacities of online data. Within the search for higher capacity devices with faster access to data, the combined interests of the vendor and user communities lead to the development and formation of two influential standards, the Peripheral Component Interface (PCI) and the Small Computer Systems Interconnect (SCSI). These standards formed the initial foundations that focused attention on the I/O challenges within the client/server architectures and enabled, more than ever before,

peripheral components to be connected to servers allowing additional devices to operate using those connections.

However, even these innovations could not effectively manage the torrent of storage demands that increased exponentially during the 1990s. The storage industry and data centers of the time searched for new ways that this overwhelming capacity of storage could be installed and accessed. This provided much of the emphasis to shift the storage connectivity paradigm from a direct connect model to a network model. With storage networking concepts emerging as a viable storage solution, this enabled two things. First, it uncovered an existing model of storage networking from the shadows of a specialized UNIX orientation, the Network Attached Storage (NAS) device. Secondly, it encouraged and spawned the integration of existing technologies to proliferate storage on its own network, thus developing into the Storage Area Network (SAN) model.

Storage networking products both in the SAN and NAS industries proliferated during the late 1990s when the technology investments were at their peak. The disparate innovation and technology du jour development this often produced has leveled into an established industry with both established and entrepreneurial players. Storage networking has crossed the product chasm and continues to evolve rapidly, although with more emphasis on data-center value and integration. Advancements are moving forward with storage initiatives such as iSCSI and connectivity through related advancements like InfiniBand and RapidIO.

Unfortunately, the value that storage networking brings to the IT community remains so diverse that it can easily become compromised by the complexities of installation and management. The most important aspect to storage networking is leveraging the value of a diverse set of solutions. I believe this will be the critical motivator in the evolution of storage networking both now and in the foreseeable future. Establishing an understanding of storage networking and identifying components of a storage networking solution and their application within the data center became key design goals for this book.

Storage Networks: The Complete Reference was designed to be a general reference for the IT community. The book was developed to address the pragmatic issues that relate to learning a new technology, understanding it, and applying it from an IT perspective. As with all reference material, it can be used in a number of ways. First, as a reference in understanding the concepts behind storage networks. Secondly, as a tool to help plan and target applications of the technology. Finally, the book is meant as an extended reference in helping your activities move beyond installation and into maintenance and management. However, it can also be used as a curriculum for classroom studies, whether employing the complete book or its parts in a formal or informal educational setting. Usage of the book depends on your existing knowledge of computers and, specifically, storage systems.

The book is intended for IT professionals who have a need to understand storage networking and its related issues from a data-center perspective. The examples in the book are composites of real-life situations and are intended to reflect major points in planning, installing, and managing storage networking in the data center.

I believe storage networking will continue to evolve into its own infrastructure, and storage networking will form the cornerstone of this migration and evolution, subsequently changing the landscape of data centers of the future.

The Complete Reference



Part I

Concepts of Storage Networking

1

The Complete Reference



Chapter 1

The Data Storage and Data Access Problem

3

Computer storage, the ubiquitous component that has invaded both our personal and working space, has taken on a life of its own with the new technology of storage networking. Like it or not, we've become a global society with a currency of information that's exchanged within today's computer networks. Nearly every aspect of our world, including the places where we live and work, are accessible through the Internet—the data from which is stored on enormous online informational databases.

Our lives revolve around computer technology. Although we don't always realize it, accessing information on a daily basis the way we do means there must be computers out there that store the data we need, making certain it's available when we need it, and ensuring the data's both accurate and up-to-date. Rapid changes within the computer networking industry have had a dynamic effect on our ability to retrieve information, and networking innovations have provided powerful tools that allow us to access data on a personal and global scale.

Both the proliferation of data and the dynamics of storing it have given way to changes in how computer storage is used and accessed. The networking innovations have driven global, local, and personal computing to greater levels of sophistication in storage technology. Because of this, we're now faced with the problems of what data to keep, how to keep it, and where to keep it. And if I do keep it, just where do I put it? We store information everywhere these days. We keep diskettes and CDs stuffed in our desks at home, collect disk and tape files at work, while countries as a whole continuously add to the mountains of archived computer tapes kept at cavernous hiding places over fear of nuclear fallout.

With so much data to store and with such global access to it, the collision between networking technology and storage innovations was inevitable. The gridlock of too much data coupled with too many requests for access has long challenged IT professionals. To storage and networking vendors, as well as computer researchers, the problem is not new. And as long as personal computing devices and corporate data centers demand greater storage capacity to offset our increasing appetite for access, the challenge will be with us. Chapter 1 will explore the data access challenges, driven by networking innovation, against data storage, exponentially increasing with innovation in media, capacity, and deployment.



The Challenge of Designing Applications

Storing and accessing data starts with the requirements of a business application. What many application designers fail to recognize are the multiple dependent data access points within an application design where data storage strategies can be key. Recognizing the function of application design in today's component-driven world is a challenging task, requiring understanding, analysis, and experience. These functions are necessary to facilitate user data within applications. The most fundamental and dependent to the

application is the data storage strategy used. Be it a file orientation or relational model, this becomes a major factor in the success of the application. (See Figure 1-1.)

In all fairness to the application designers and product developers, the choice of database is really very limited. Most designs just note the type of database or databases required, be it relational or non-relational. This decision in many cases is made from economic and existing infrastructure factors. For example, how many times does an application come online using a database purely because that's the existing database of choice for the enterprise? In other cases, applications may be implemented using file systems, when they were actually designed to leverage the relational operations of an RDBMS.

Whether intelligently chosen or selected by default, the database choice leads directly to the overall interaction the application will have with storage. The application utilizes, and is dependent upon, major subsystems, such as memory management (RAM and system cache), file systems, and relational and non-relational database systems. Each of these factors affects the storage environment in different ways. An application with no regard for these factors results in application functionality that doesn't account for the availability, limitations/advantages, and growth of storage resources. Most business applications simply don't take into consideration the complexity of storage performance factors in their design activities. Consequently, a non-linear performance-scaling factor is inherent in business applications, regardless of whether the application is an expensive packaged product or an internally developed IT utility program. (See Figure 1-2.)

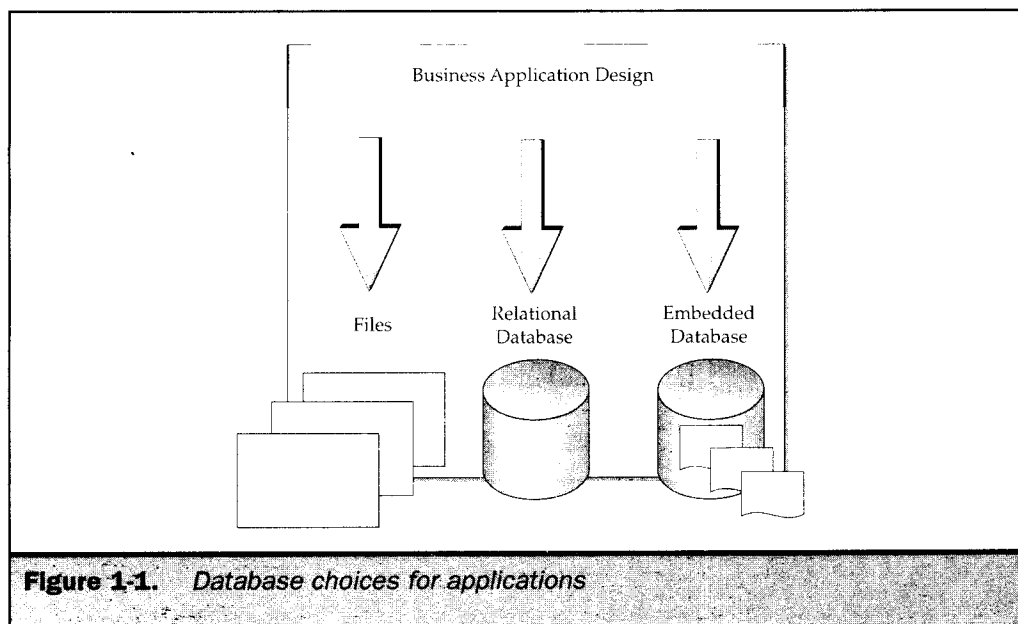


Figure 1-1. Database choices for applications

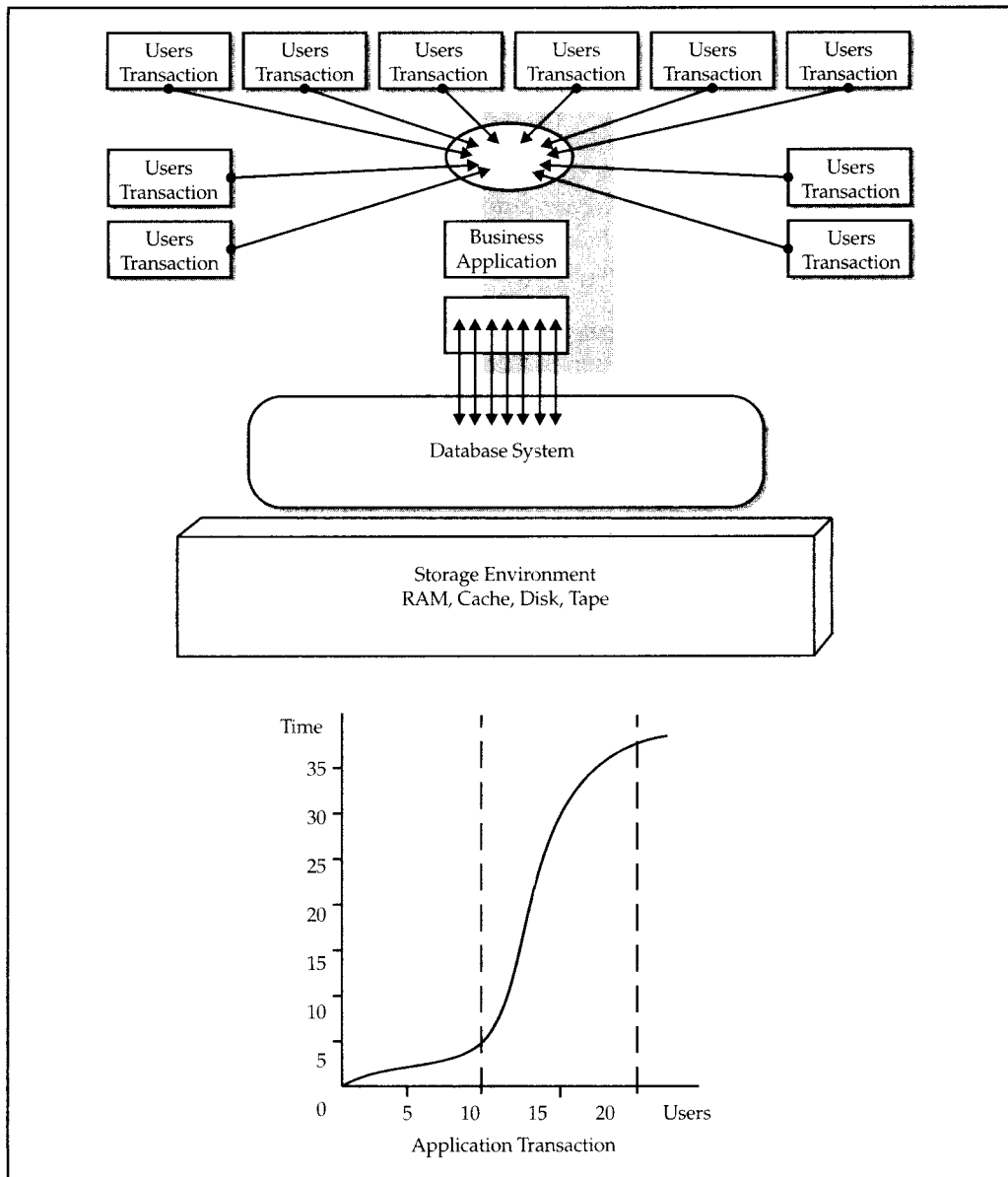


Figure 1-2. Components that effect application performance

Non-Linear Performance in Applications

The major factors influencing non-linear performance are twofold. First is the availability of sufficient online storage capacity for application data coupled with adequate temporary storage resources, including RAM and cache storage for processing application transactions. Second is the number of users who will interact with the application and thus access the online storage for application data retrieval and storage of new data. With this condition is the utilization of the temporary online storage resources (over and above the RAM and system cache required), used by the application to process the number of planned transactions in a timely manner. We'll examine each of these factors in detail.

First, let's look at the availability of online storage. Certainly, if users are going to interact with an application, the information associated with the application needs to be accessible in real time. Online storage is the mechanism that allows this to happen. As seen in Figure 1-3, the amount of online storage required needs to account for sufficient space for existing user data, data the application requires, and unused space for expanding the user data with minimal disruption to the application's operation.

However, as users of the application submit transactions, there must be locations where the application can temporarily store information prior to the complete execution of the transactions—successful or otherwise. Given that online storage is the permanent record for the application, the use of RAM (the computer's memory resource) and system or disk cache (a temporary memory resource associated with the CPU component or online storage component, also depicted in Figure 1-3) is used. Because of disk cache's

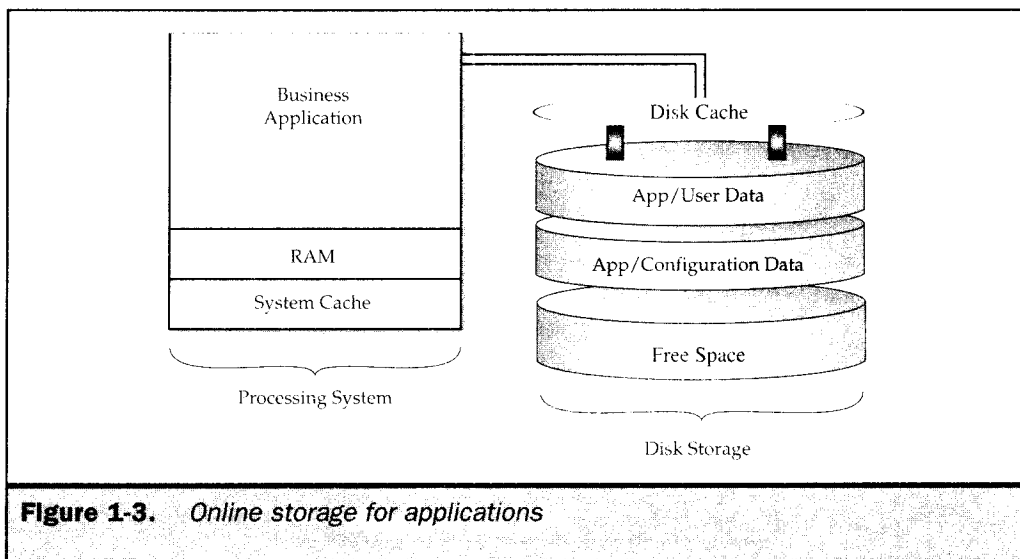


Figure 1-3. Online storage for applications

high speed and capability to place significant amounts of application data temporarily until the complete execution of the transaction, this storage strategy not only provides the fast execution of data, but also the necessary recovery information in case of system outage, error, or early termination of the application transaction.

Second, among the factors influencing non-linear performance, is the number of users accessing the application or planning data access. Indicated in Figure 1-4, the number of

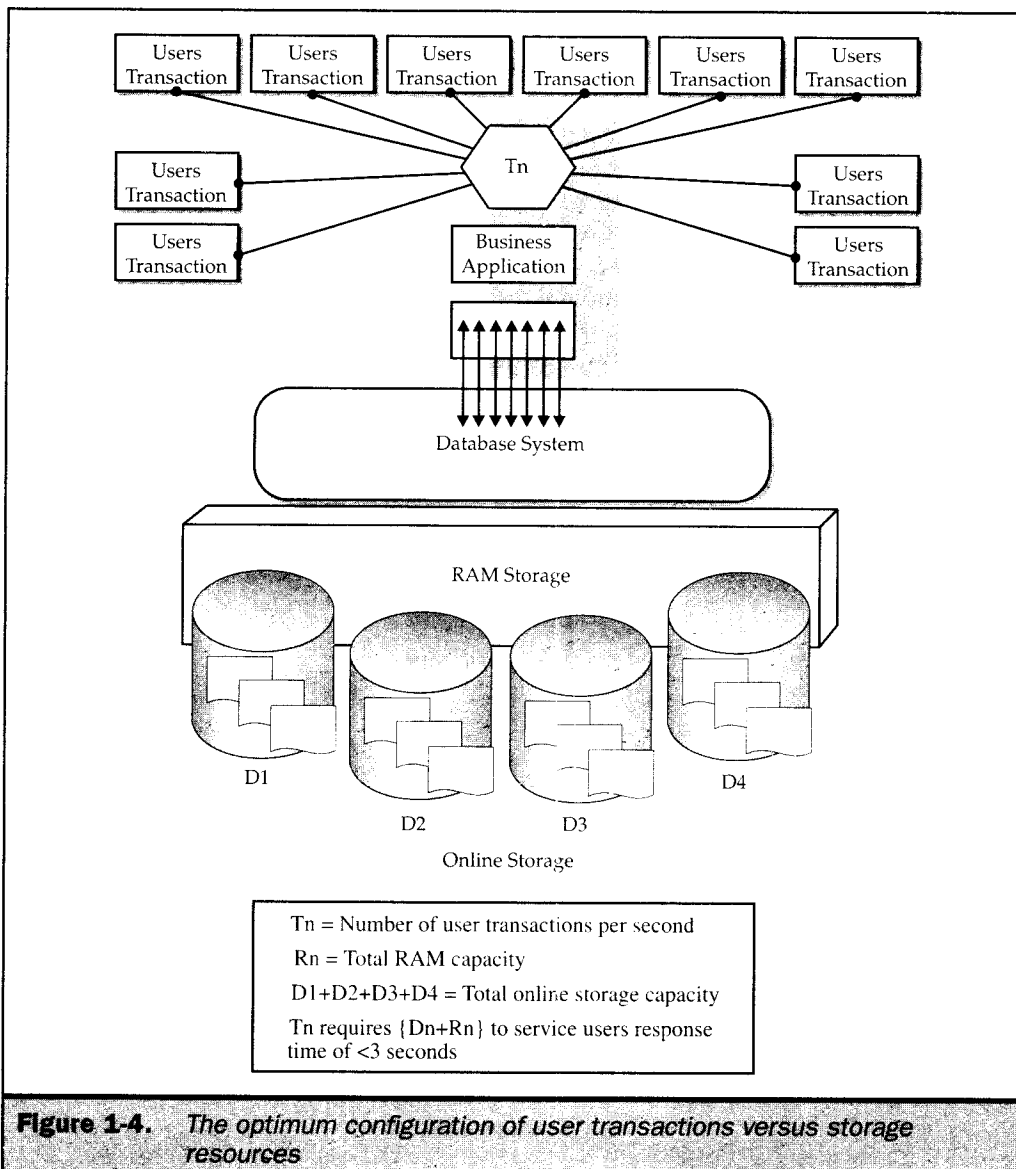


Figure 1-4. The optimum configuration of user transactions versus storage resources

user transactions accessing storage resources will correspond linearly to the amount of time each transaction takes to complete (provided optimum resources are configured). Therefore, configurations can be developed so a fixed number of users will be serviced within an acceptable response-time window (given optimum storage resources). Although these optimum configurations depend on server computing capacity and adequate network resources, the transactional demands of the application rely on the storage infrastructure. (See Figure 1-5.)

It's unlikely that application designers will ever fully develop systems that take into account the necessary optimum storage configurations. The use of pre-existing components for database and file access will continue, further removing the application from the storage infrastructure. These tendencies have placed additional demands on storage strategies in order that the balance between application and storage resources can be maintained. To address the data storage/data access problem, designers have been forced to think outside of the box.

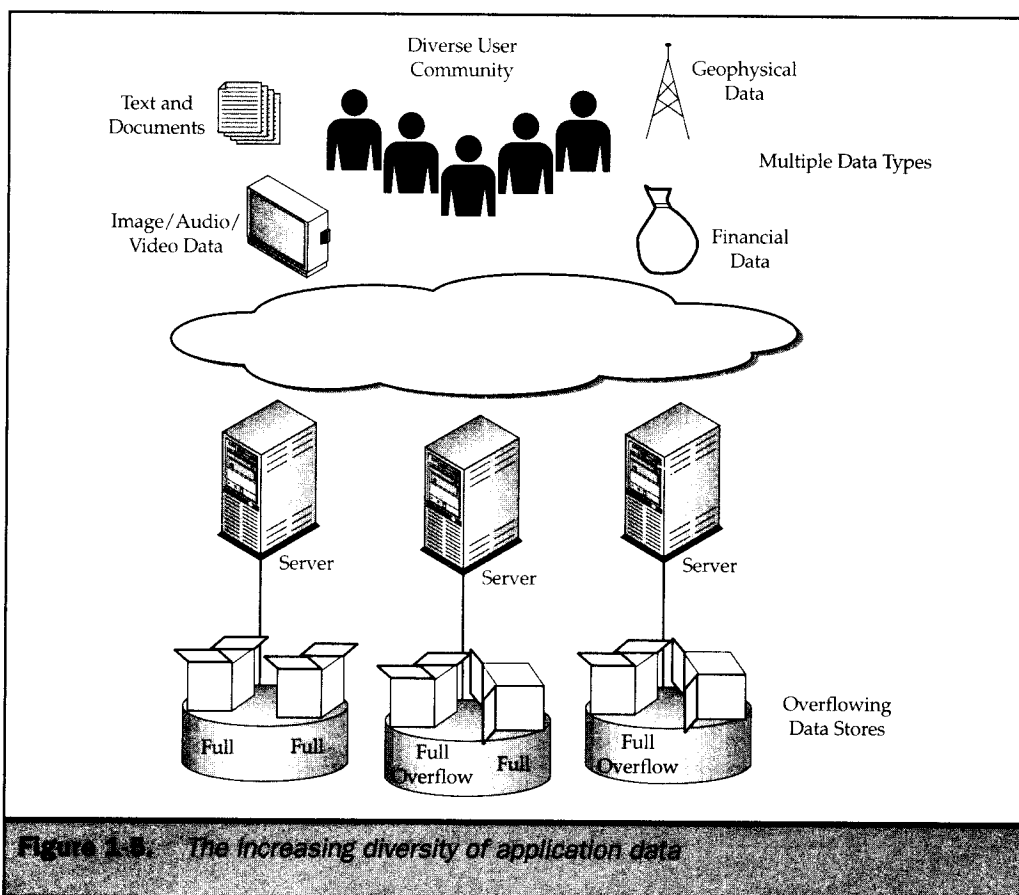


Figure 1-5. The increasing diversity of application data

The Client/Server Storage Model

Our storage configurations today reflect the characteristics of the client/server model of distributed computing. Evolving out of the centralized mainframe era, the first elementary configurations of networking computers required a central computer for logging in to the network. This demanded that one computer within a network be designated, and that the necessary information be stored and processed on that computer in order for other users to access the network. Thus, the network server became the location where (on its hard drive) network information was stored. When users logged in from their client computers, they first accessed the information on the network server. Figure 1-6 shows the structure most storage configurations use today: the client/server storage model.

The client/server storage model provides data storage capabilities for the server component as well as storage for the network clients. Coupled with online storage, the servers became larger machines with increased RAM and cache to handle the multiple requests for network access demanded by users. The online storage devices also became more robust. With larger capacities, its own caching mechanisms, and greater external connectivity for expansion, it enabled faster data access. However, servers quickly

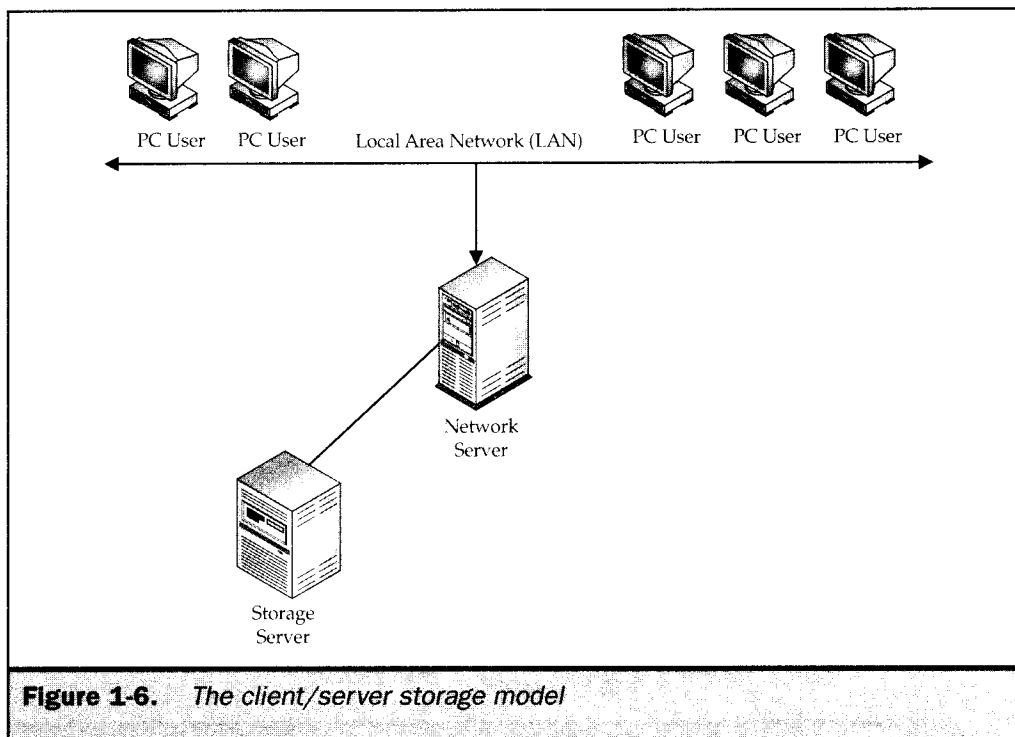


Figure 1-6. *The client/server storage model*

reached their maximum performance capacities as the client user community demand grew faster than server capabilities. After a while, the optimum storage configuration became almost impossible to achieve given server performance limitations.

Along with the demand for client network connectivity, sharing information on the server increased the need for capacity, requiring networks to use multiple servers. This was not due so much to server capacity but that online storage needed to be large enough to handle the growing amount of information. So, as demand for online information increased so did the need for storage resources, which meant the number of servers had to increase as well.

This quickly grew into the world of server specialization. The network servers had to handle just the amount of work necessary to log clients into and out of the network, keep their profiles, and manage network resources. Client profiles and shared information were now being kept on their own file servers. The demand for online storage space and multiple access by clients required that multiple servers be deployed to handle the load. Finally, as the sophistication of the centralized mainframe computers was downsized, the capability to house larger and larger databases demanded the deployment of the database server.

The database server continues to be one of the main drivers that push the client/server storage model into new and expanded storage solutions. Why? Initially, this was due to size. The amount of information stored within the databases, driven by user demands, quickly surpassed the capacities of the largest online storage providers. In addition, the architecture of the relational model of databases (from products like Oracle, Sybase, IBM's DB2) had a tremendous amount of overhead to provide the extremely attractive functionality. Consequently, the user data only utilized half of the space needed, with the database occupying the remaining half.

As the relational database model became pervasive, the amount of databases within the network grew exponentially. This required that many databases become derivatives of other databases, and be replicated to other database servers within the enterprise. These activities began to drive the storage growth rates to extremely high levels—on average, 75 percent, but greater than 100 percent occurred in some cases.

The database server model also required that servers become more robust and powerful. Servers supporting databases evolved into multiple CPUs, larger amounts of RAM, levels of system cache, and the capability to have several paths to the external storage resources. Essentially, these became configured exclusively as storage servers, with RAM size, cache, and disk storage capacities being key configuration elements. However, it also required system and storage planners to provide additional network paths to online storage resources in order to keep up with user demand. Figure 1-7 depicts the level of sophistication required as database servers became the dominant storage consumers.

As data grows, storage solutions with the client/server storage model continue to be problematic. Even the multiple CPU servers and multigigabyte storage capacities could not ensure the server's capability to provide optimum client access. Limits

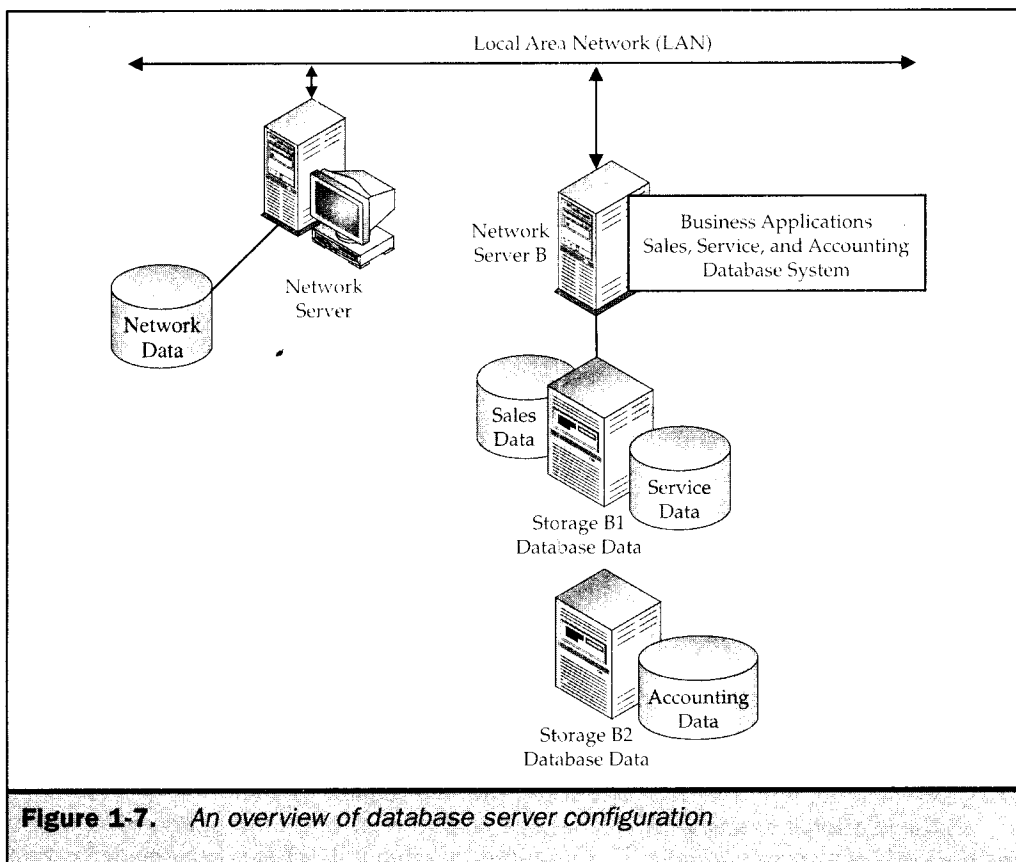
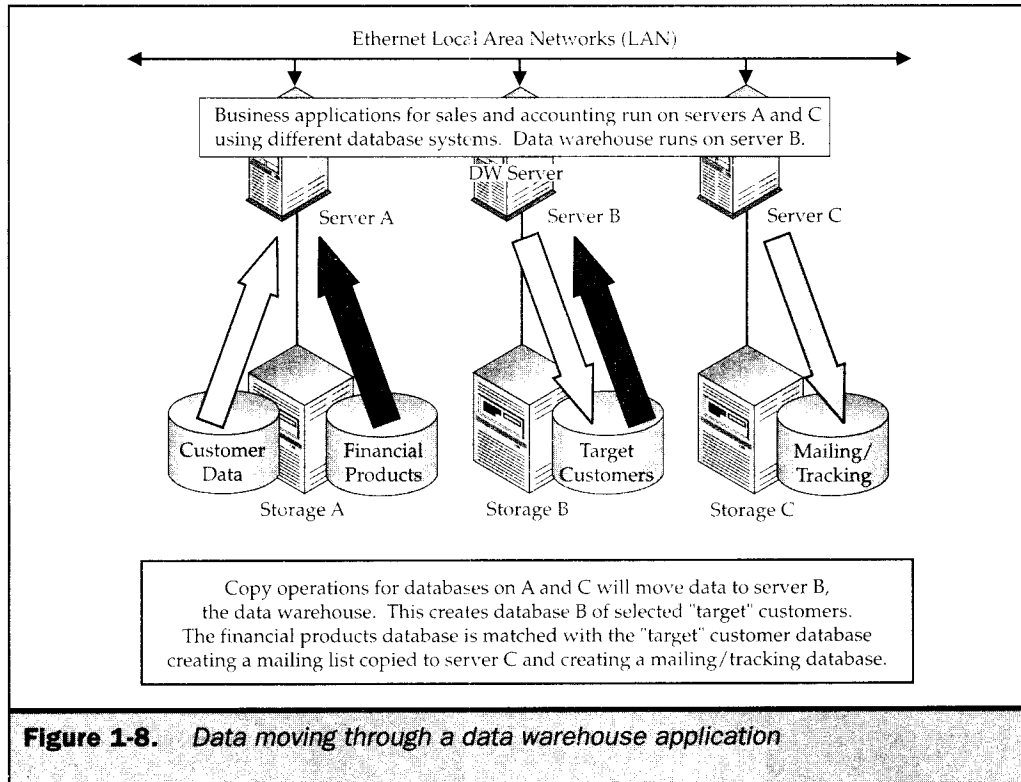


Figure 1-7. An overview of database server configuration

to the size of database systems, increasing content of database transactions, and the advent of new datacentric applications required more sophistication within the storage systems. However, it also demanded that more servers be used to provide access to application information. As a result, growth rates began to show a varied and asymmetrical growth trend: some servers were deployed with more storage than required, wasting space; while other servers continued to be split because their capacity was exceeded, causing application outages due to space shortages.

An example is the exponential growth factors for storage supporting the data warehouse and data mart application segments. Typical of this is a target marketing application that ensures customers with good credit are selected and matched with offerings for financial investments. These matches are then stored and targeted for mailings in which responses are tracked. Figure 1-8 illustrates the operational complexities of storage exploitation with these environments.



The actual data traverses from online storage "A" on server A, where the customer activity and tracking database resides, to populate the data warehouse server online storage "B" on server DW. This selection of good customers from server A populates the database on server DW's online storage "B" where it is matched with financial investment product offerings also stored on server A. Subsequently, the aggregate data on C is developed from combining good customers on server B to form a mailing and tracking list stored on server DW's online storage "C."

As we have described in our example, data moves from server to server in the client/server storage model. The capability to store multiple types of data throughout the network provides the foundation for selecting, comparing, and utilizing this data for business purposes throughout the enterprise. However, this comes at a price since databases are costly in terms of large external storage and server resources, which they require to both process data and communicate with the network. Moreover, the capability to maintain these environments—database systems, more sophisticated hardware, and storage infrastructure—requires additional technicians and programmers.

As data grows, so does user access. The problem of access has to do with understanding the challenges of the client/server storage model—the biggest challenge being the Internet. The amount of data stored from the Internet has stressed the most comprehensive systems and pushed centralized mainframes back into the limelight purely on their strength in supporting thousands of online users. However, largely due to the economics of client/server systems, Internet infrastructures continue to be based on the client/server storage model.

As web servers began to grow due to the ease of deploying web pages, not to mention the capability to connect to an existing networking structure (the Internet, for instance), the amount of users able to connect to a web server became limited only by addressing and linking facilities. Given the client/server connectivity, servers on the Internet essentially had no limitations to access, other than their popularity and linking capability. Because of this, an even more aggressive move was required to populate multiple servers and specialize their uses. The example shown in Figure 1-9 shows how web servers specialize in user access, user profiles and accounts, web pages, and user mail.

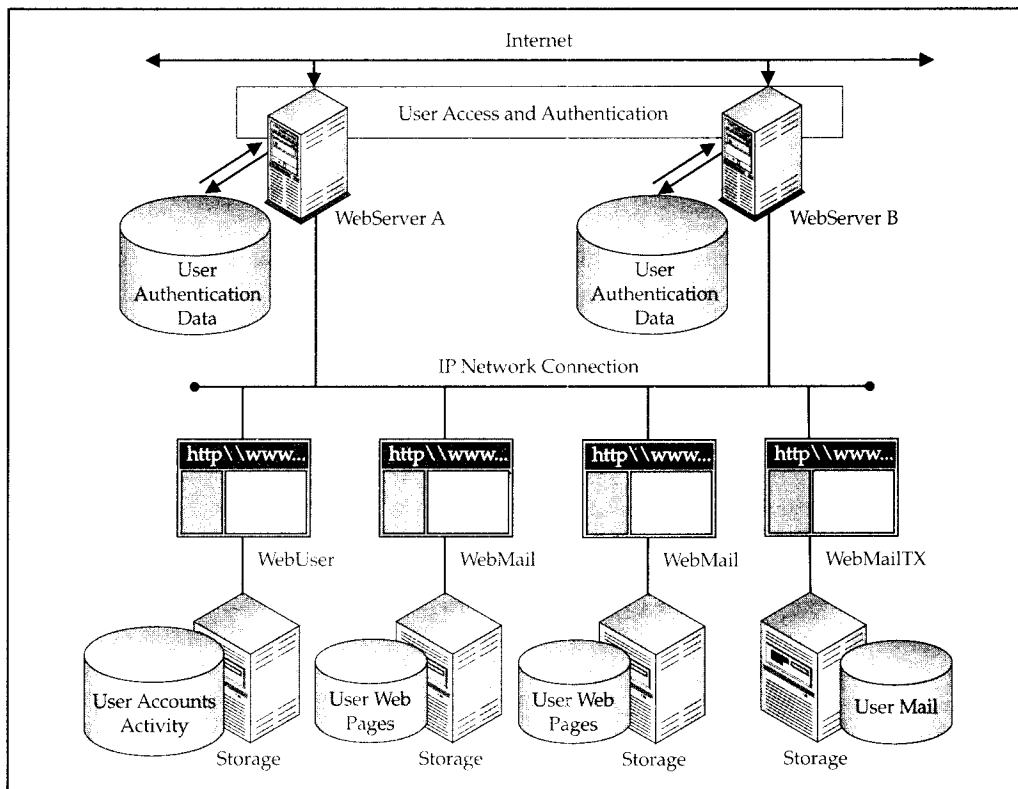


Figure 1-9. An overview of web server storage

and mail. Web servers are also replicated to balance the demands of users accessing particular sites.

If we examine the performance and growth factors for storage supporting the web server applications, we find data moving in a fairly predictable way. A typical example is the infrastructure of an Internet service provider (ISP). Although they support multiple web-based applications, their main service is to provide personal users with access to the Internet. Therefore, they store two fundamental items, user profiles and mail, as well as links to other servers on the Internet. Figure 1-9 illustrates the operational simplicity of storage exploitation within these environments.

The actual data traverses from online storage on web server A, where the customer authentication and access takes place, then moves to online storage on WebUser servers where profiles, activity tracking, and home pages are stored. Essentially, the user waits here after logging in, until they issue a request. If the request is to check e-mail, the WebUser transfers the request to the WebMail server where the request is placed in an index and transferred to the location of the user's mail. In this example, we're using WebMail TX.

However, if the user issues a request to access another URL in order to browse an outside web page, the WebUser server will send the request to web server A or B. Either of these servers will transmit the request through the Internet, locate the outside server, bring back the web page or pages, and notify the WebUser server who in turn will transmit the web page(s) to the customer's client PC.

Although Figure 1-6 is a simple example of an ISP's web server configuration, the process becomes convoluted upon much interaction with the ISP's storage infrastructure. We can estimate the scope of this infrastructure by multiplying the number of customers by one million. If each customer stores 8MB of data within their profiles, e-mail, and activity tracking, this becomes 8 terabytes, or 8 million MBs, of necessary online data storage. In estimating the scope of interaction (or data access), each customer issues, on average, 20 requests for data with most being outside the ISP infrastructure. Each request requiring, on average, 500KB of data to be transferred, therefore becomes 10MB of data transferred. Multiplying this by one million customers equals 10 terabytes of data transferred through the ISP storage infrastructure every half hour.

The capability to maintain these environments, like the database server environments with database systems, multiple servers, and exponentially expanding storage infrastructure, requires an ever-increasing number of skilled technicians and programmers.

Reaching the Client/Server Storage Limitations

The preceding examples of datacentric applications, data warehouses, and ISPs demonstrate the increasing limitations of the client/server storage model. As the demand for data and online storage continues, capacities and functionality must grow to meet these challenges. Although disk drives, are becoming physically smaller and are holding increased amounts of data, and temporary storage devices such as caching have advanced, they cannot overcome the I/O limitations of the client/server architecture.

We are left asking ourselves: Is it necessary to go through the server just to get to the data? Why can't we maximize all the online storage we have? The amount of servers we have has far exceeded our capability to manage the attached storage. Isn't there a way to share data among servers rather than duplicate it throughout a company?

In addition, for every byte stored on disk there is the possibility that someone will access it. More and more users are coming online through Internet-based capabilities. Increased functionality and access to a diversity of data has created an informational workplace. Our own individual demands to store more and more data are pushing the limits of computer networks. It will be an increasing challenge to accommodate a global online community with client/server storage configurations.

The Boot Disk of Innovation

Data flow challenges have historically been the "boot disk" of innovation. The data storage/data access problems have prompted the industry to bootstrap itself with creative solutions to respond to the limitations of the client/server storage model. In 1995, in what began as efforts to decouple the storage component from the computing and connectivity elements within a computer system, became a specialized architecture that supported a significantly dense storage element (size, for example) with faster connectivity to the outside world. Combining elements from the storage world with innovation from networking technologies provided the first glimpse into how the industry would address the data storage/data access phenomenon. The collaborative evolution of these two distinct entities, storage and networking, formed the genesis of what we know today as storage networking.

As with most "out of the box" innovations, storage networking combines aspects of previous ideas, technologies, and solutions and applies them in a totally different manner. In the case of storage networking, it changes the traditional storage paradigm by moving storage from a "direct" connection to the server to a "Network" connection. This design places storage directly on the network. This dynamic change to the I/O capabilities of servers by decoupling the storage connectivity provides a basis for dealing with the non-linear performance factors of applications. It also sets the foundations for highly scalable storage infrastructures that can handle larger data access tasks, share data across servers, and facilitate the management of larger online storage capacities. All of which are foundations for building a new storage infrastructure and provisioning a comprehensive data storage/data access strategy.

However, this architectural change comes with costs. It requires rethinking how applications are deployed and how the existing client/server storage model should evolve, and, how to develop new storage infrastructures. This will require a revised view of IT storage infrastructure strategies. Characterized by struggles over size and access, IT professionals and storage vendors have moved to the front lines of the battle.