

# NETWORK ATTACHED STORAGE ARCHITECTURE

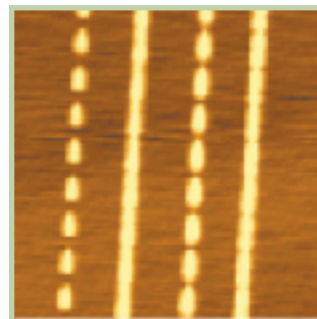
*In our increasingly Internet-dependent business and computing environment, network storage is the computer.*

THE GROWING MARKET FOR NETWORKED STORAGE IS A RESULT OF THE EXPLODING DEMAND FOR STORAGE CAPACITY IN OUR INCREASINGLY INTERNET-DEPENDENT WORLD AND ITS TIGHT LABOR MARKET. STORAGE AREA NETWORKS (SAN) AND NETWORK ATTACHED STORAGE (NAS) ARE TWO PROVEN APPROACHES TO NETWORKING STORAGE. TECHNICALLY, INCLUDING

A FILE SYSTEM IN A STORAGE SUBSYSTEM DIFFERENTIATES NAS, WHICH HAS ONE, FROM SAN, WHICH DOESN'T. IN PRACTICE, HOWEVER, IT IS OFTEN NAS'S CLOSE ASSOCIATION WITH ETHERNET NETWORK HARDWARE AND

SAN with Fibre Channel network hardware that has a greater effect on a user's purchasing decisions. This article is about how emerging technology may blur the network-centric distinction between NAS and SAN. For example, the decreasing specialization of SAN protocols promises SAN-like devices on Ethernet network hardware. Alternatively, the increasing specialization of NAS systems may embed much of the file system into storage devices. For users, it is increasingly worthwhile to investigate networked storage core and emerging technologies.

Today, bits stored online on magnetic disks are so inexpensive that users are finding new, previously unaffordable, uses for storage. At Dataquest's Storage2000 conference last June in Orlando, Fla., IBM



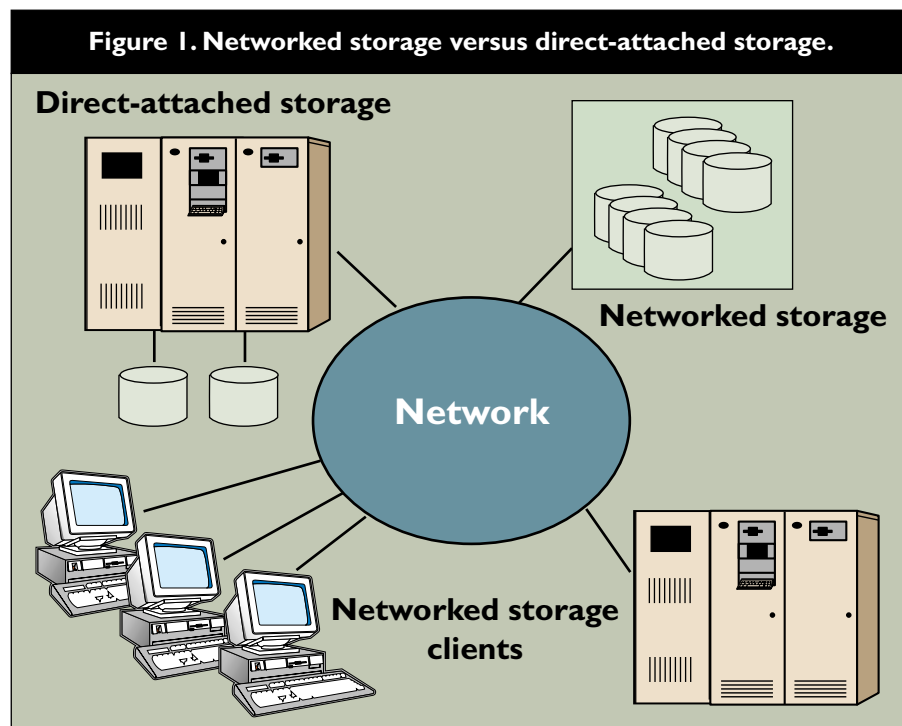
reported that online disk storage is now significantly cheaper than paper or film, the dominant traditional information storage media. Not surprisingly, users are adding storage capacity at about 100% per year. Moreover, the rapid growth of e-commerce, with its huge global customer base and easy-to-use, online transactions, has introduced new market requirements, including bursty, unpredictable spurts in capacity, that demand vendors minimize the time from a user's order to installation of new storage.

GARTH A. GIBSON AND RODNEY VAN METER

The rapid increase in the need for online capacity fuels other business technology trends. First, capital investment in storage is now more than 50% of all capital investments in corporate data centers. Industry analysts predict this percentage will reach 75% by 2003. Second, personnel costs for storage management (for, say, tuning performance and maintaining backups) now dominate capital costs over the equipment's useful lifetime. Vendors estimate this recurring cost to be as much as \$300 per gigabyte per year; that is, each year's recurring cost is comparable to, and often exceeds, its one-time capital-cost counterpart. Coupled with the shortage of information technology professionals and the bursty, unpredictable capacity needs of e-commerce, it is therefore not surprising that a new market has emerged for data-center outsourcing for both complete applications (application service providers) and storage systems only (storage service providers). Third, the increasing cost of storage management, coupled with the continuing decline in the cost of storage capacity, has prompted analysts to predict that by 2005 the primary medium for storage backup will be online hard disks.

Various system properties are improved by separating storage from application servers and client machines and locating it on the other side of a scalable networking infrastructure (see Figure 1). Networked storage reduces wasted capacity, the time to deploy new storage, and backup inconveniences; it also simplifies storage management, increases data availability, and enables the sharing of data among clients.

It reduces wasted capacity by pooling devices and consolidating unused capacity formerly spread over many directly attached storage servers. It reduces the time needed to deploy new storage, because client software is designed to tolerate dynamic changes in network resources but not the changing of local storage configurations while the client is operating. Data backup traditionally occupies application data servers and hosts for much of the night, a significant inconvenience for global and always-open organizations.

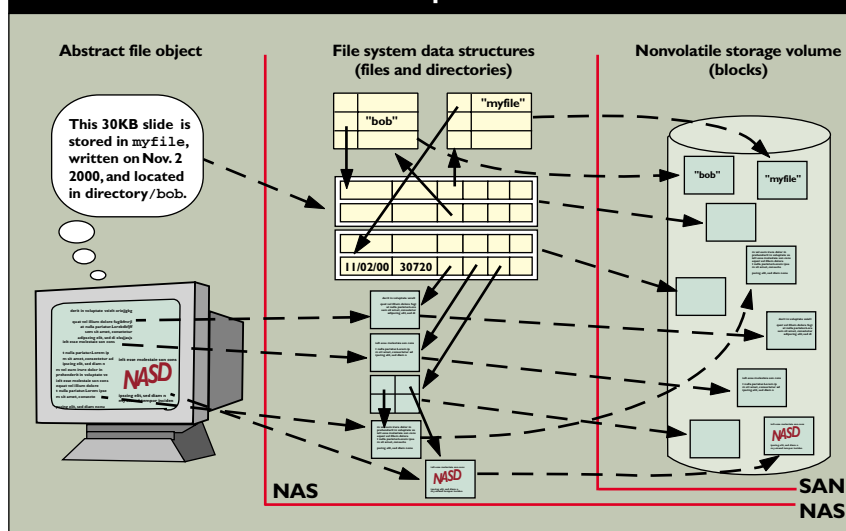


With networked storage, backup can be made less inconvenient, because data can be transferred to offline tape storage when the devices are not busy—day or night—without application-server involvement. Networked storage also simplifies storage management by centralizing storage under a consolidated manager interface that is increasingly Web-based, storage-specific, and easy to use.

Inherent availability, at least in systems in which all components are provided by the same or cooperating vendors, is improved, because all hardware and software in a networked storage system is specifically developed and tested to run together. Traditional servers are more likely to contain unrelated hardware and software added and configured by users. Such user additions can be destabilizing, because all components are unlikely to have been sufficiently tested together, resulting in more frequent crashes. Finally, the sharing of data among clients is improved, because all network clients can access the same networked storage.

The principal disadvantages of networked storage are due to the increased complexity of systems spread across a network. More machines have to function correctly to get work done; network protocol processing is more expensive than local hardware device access; and data integrity and privacy are more vulnerable to malicious users of the network. While network storage technology must address these disadvantages, all distributed and Internet applications share them; hopefully, network storage technology will share their

**Figure 2. Internal storage interfaces and data structures and their relationships with one another.**



solutions, as well as their problems.

Most networked storage systems fall into one of two technology families: NAS systems (such as the F700 file-server systems from Network Appliance of Sunnyvale, Calif.) typically accessed via Ethernet networks; and SAN systems (such as the Symmetrix disk array from EMC of Hopkinton, Mass.) typically accessed via Fibre Channel networks [1]. Both NAS and SAN storage architectures provide consolidation, rapid deployment, central management, more convenient backup, high availability, and, to varying degrees, data sharing. It is therefore no surprise that an IT executive might view NAS and SAN as solutions to the same problems and the selection of networking infrastructure as the principle differentiator among them. The technology trends we discuss here are likely to blur this simplistic, network-centric differentiation between NAS and SAN, so we begin with the principle technological difference.

### NAS and SAN Interfaces

Technologically, NAS and SAN systems offer different abstractions to the software using them [1, 2]. At its simplest, the difference between a NAS storage abstraction and a SAN storage abstraction is that NAS systems offer file system functionality to their clients and SAN systems do not.

SAN systems provide a simple, untyped, fixed-size (block), memory-like interface (such as `get block`, `set block`) for manipulating nonvolatile magnetic media. From the perspective of a datapath abstraction, there is little functional difference between the interfaces of a SAN system and a traditional attached disk. Even though a SAN network makes it possible for multiple clients to access the

same storage device directly, this type of sharing requires coordination not presented by the SAN interface to ensure that concurrent access is synchronized.

Here, we group together the storage and the storage-area network hardware to which the storage is attached, referring to them collectively as the SAN system. In this sense, a Fibre Channel disk is a SAN device. A more narrow interpretation of SAN includes only Fibre Channel hubs, switches, and routers. In the following sections, we use the term “SAN systems” to mean SAN-attached storage systems.

NAS systems provide a richer, typed, variable-size (file), hierarchical interface (including create or delete file, open or close file, read or write file subset, get or set file attribute, create or delete directory, insert or remove link between directory and file or directory, and lookup file name). NAS systems internally interface with nonvolatile magnetic media, usually through a SAN-like interface to ensure data reliability. From the perspective of a datapath abstraction, there is little functional difference between the interfaces of a NAS system and those of a traditional local file system.

Figure 2 outlines the relationship between the NAS and SAN interface abstractions, using as an example a single user file containing a presentation slide image. On the left, a user's document is preserved between editing sessions in a file whose local name is “myfile.” Stored in a file system, this file has been located in a collection of files (a directory), called “bob” in the example. Uniquely naming the user's file requires listing its name and its ancestor directories' names, “/bob/myfile” in the example. As a side-effect of its creation, “myfile” is labeled with such attributes as the date of its creation and the amount of storage used by its contents. In the example, these attributes are “November 2, 2000” and “30 KB,” respectively.

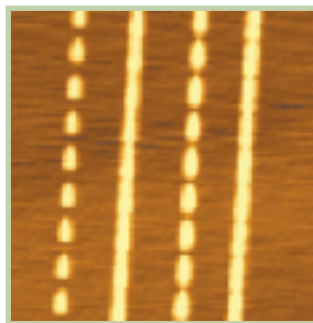
The diagram in the middle of Figure 2 outlines how “myfile” might be represented in the data structure of a Unix-like file system [8]. There are four different types of uses for fixed-size blocks in this data structure: user data, directory entries, file descriptors, and indirect pointers. All data structures other than the user data are known as metadata and must be con-

sulted as a file system executes user requests on data files.

In this example, four of the boxes at the bottom are the four data blocks needed to store the contents of “myfile.” The two boxes at the top of the diagram are blocks containing directories. A directory associates human-readable file names with pointers to the corresponding files’ descriptors. The two blocks in the middle of the diagram contain the file system’s table of file descriptors. A file descriptor is a fixed-size record containing in its hundreds of bytes a few attributes of the file, a few (direct) pointers to the first few data blocks of the file, and a few pointers to “indirect” blocks.

On the right-hand side of Figure 2 is an illustration of a SAN system’s internal components. Such systems

Table 1. Scaling size stresses implementation mechanisms.	
What is being scaled	What mechanisms are stressed?
number of client and server nodes	resource discovery; network bandwidth and congestion control; addressing
distance	network congestion and flow control; latency and round-trip dependencies; resource discovery; security; network routing
aggregate and individual bandwidth	interconnect technology; protocol processing in client and server OS
number and size of files	application addressing; file metadata management
directory size and tree depth	file metadata management; round-trip time for repeated name lookup



**NETWORKED STORAGE REDUCES WASTED CAPACITY, THE TIME TO DEPLOY NEW STORAGE, AND BACKUP INCONVENIENCES; IT ALSO SIMPLIFIES MANAGEMENT, INCREASES DATA AVAILABILITY, AND ENABLES THE SHARING OF DATA AMONG CLIENTS.**

contain an undifferentiated set of fixed-size blocks named according to their position in a list of all such blocks. This set of fixed-size blocks is stored on non-volatile storage (typically, magnetic-disk media), such that a sequential walk through the list of blocks exploits a device’s maximum data rate. Dotted lines represent the relationship between the file system’s data and metadata structures and their persistent storage on the blocks of the SAN system.

While NAS and SAN interfaces are functionally similar to traditional file systems and storage device interfaces, NAS and SAN offer much more manageable storage to a data center’s staff. SAN devices, whether internally or with SAN-specific software, typically represent multiple disks as if there were only one disk (a virtual disk volume), simplifying space management, and transparently maintain redundant data (or redundant array of inexpensive disks, RAID, encodings), thus increasing availability and reliability. NAS systems inherit these advantages because they are usually built on SAN storage. Moreover, new NAS systems can be installed and configured without interrupting the execution of client machines. And NAS users on different machines can see and share the

same collection of files without special effort. These manageability advantages of NAS and SAN systems are compelling to those responsible for an organization’s information technology, given the general scarcity of storage administration professionals.

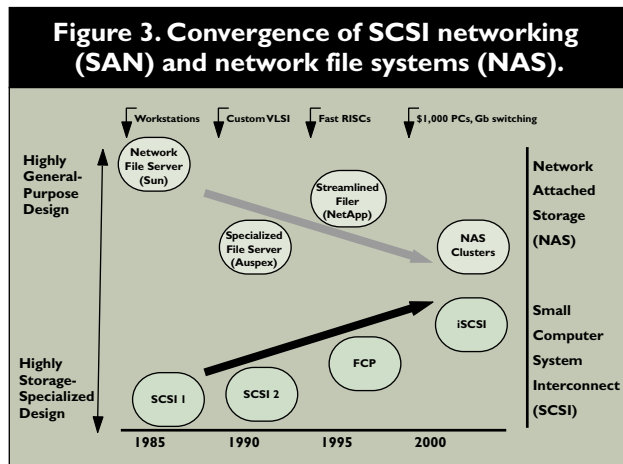
### Requirements for Emerging Systems

Future network storage systems must provide features to meet all existing requirements, including resource consolidation, rapid deployment, central management, convenient backup, high availability, and data sharing, as well as the following emerging requirements.

*Geographic separation of system components.* Because online commerce is increasingly global and competitive, remote data must be continuously available, and all data must have remote copies updated frequently to protect against regional disasters. Moreover, with the Internet infrastructure’s bandwidth growing at upward of 300% per year, employing the Internet in an organization’s own internal network is increasingly affordable and effective.

*Increasing risk of unauthorized access to storage.* Security is an increasingly critical storage property as





online commerce becomes important and as electronic crime (hacking) becomes increasingly sophisticated and common. Moreover, storage-system interconnects, including most SANs today, were originally designed as extensions of the internal buses of their hosts; their security provisions are limited or nonexistent. Extended buses may be easily secured from hostile traffic through physical means, but they are also limited in geographic distribution and the number of attached devices they can support. Linking extended buses to other networks, now possible with storage interconnects like Fibre Channel, greatly weakens these physical controls. Mechanisms for restricting access to network-storage servers are even more important on the Internet than in standalone Fibre Channel networks.

**Need for performance to scale with capacity.** Performance, measured either as accesses per second or megabytes per second, needs to scale with storage capacity to accommodate the increasing power and number of client machines, as well as the increasing size of datasets (for applications manipulating such data as sensor traces, transaction records, still images, and video).

In general, the increasing scale and scope of the use of storage systems drives these emerging requirements. See Table 1 for the system parameters and implementation mechanisms most likely to be stressed beyond their design goals by the increasing scale and scope of how these systems are used.

The storage systems designed to meet these requirements are likely to be structured around their answers to a few critical architectural questions: What is the storage abstraction for the network interface? How many network crossings per unit of application work are required? What are the bottleneck functions? Do control and data travel the same path? How is clustering used? What parts of the system are trusted?

NAS places a network between client and file system, and SAN places a network between the file system and storage media. Other options for the abstraction of a networked storage interface might be to place a network between client and application, as is done by such application servers as database engines, or between the file-system directory functions and the files themselves, as in the object-based storage devices, such as the Network-Attached Secure Disks (NASD) system, discussed later. Moreover, the attributes associated with stored blocks, objects, or files could be much richer; in addition to recording dates, sizes, and permissions, attributes could be used to set performance goals, trigger backups, or coordinate shared access with locks and leases.

The number of network crossings needed to complete a unit of application work is critical when components of the system may be separated by large geographic distances. Block-level abstractions offered by SAN systems send more storage requests across the network to do the same work, because client-based file systems may have to sequentially fetch multiple metadata blocks before they are able to fetch data blocks. In contrast, NAS systems contain the file system that interprets metadata, so they do not send as much metadata across the network.

Traditional NAS and SAN systems have many disks per storage-controller processor to amortize the cost of the storage controller. This architecture renders the controller a bottleneck, because today's disks move data efficiently (compared to general-purpose processors), and file-system command processing uses relatively large numbers of server processor cycles. One strategy for avoiding controller bottlenecks is to separate control and datapaths. This classic direct-memory-access approach allows the datapath to be specialized for speed. Another strategy for avoiding bottlenecks is to "parallelize" the bottleneck controller component into a coordinated, load-balanced cluster of controllers.

Deciding what components and which network messages to trust is the core of any security architecture. Traditionally, SAN storage trusts everything attached to and received over its network. NAS systems have traditionally trusted less, mainly the operating systems of their clients but not the users of these clients. Today's security techniques, such as virtual private networks like IPSec, firewalls, and Fibre Channel zoning, allow SAN systems to reduce their trust domains to mainly their clients' operating systems by limiting traffic on the network to only participating systems. Cryptographically authenticated connections between end users and servers would further improve the ability to discriminate among authorized and

unauthorized requests on shared networks.

## Converging of NAS and SAN

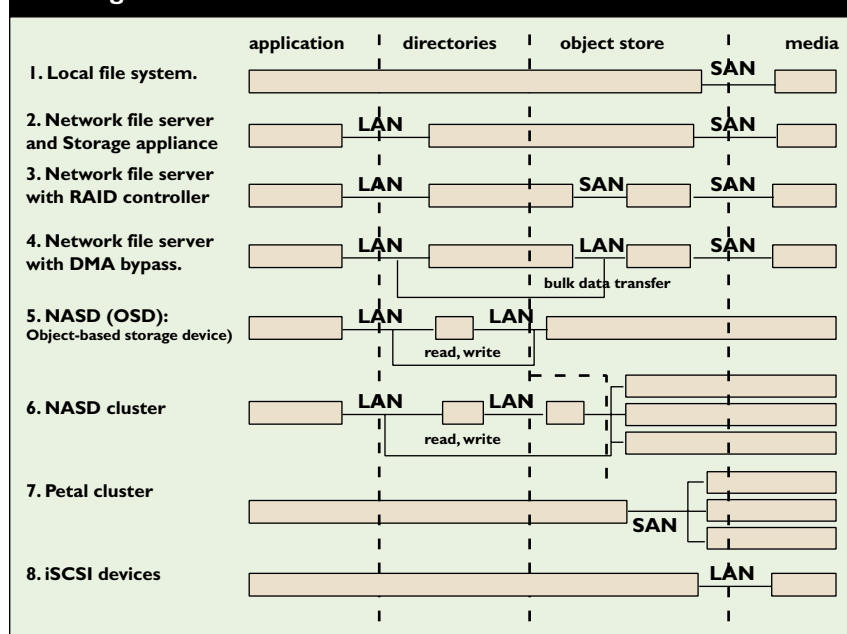
Although the placement of file system functions is an important difference in the interface abstraction of NAS and SAN systems, in other ways these technologies are becoming more similar [10]. Discussed earlier was how they provide solutions to the same set of customer problems, making it reasonable for an executive responsible for information technology to view them as interchangeable alternatives. However, they are also converging in another way: The degree to which their implementations are specialized for storage is increasing in NAS systems and decreasing in SAN systems. Specialization in hardware, software, or networking often yields more efficient use of resources (including memory bandwidth, die space, and manufacturing costs). However, specialization is costly and restricts the market size from which its development costs can be recovered. Generalized components benefit from amortizing development costs over much larger markets, so the rate of improvement is often much faster. Moreover, the use of generalized components creates opportunities for leaps in technology capability, because it is sometimes possible to reuse complete and complex solutions developed for other problems.

Figure 3 reflects the specialization trends in NAS and SAN systems, using the Small Computer Systems Interface (SCSI)—the dominant SAN command protocol—to illustrate SAN specialization.

**More specialized NAS systems.** Network attached storage, originally known as network file service, was developed as a side effect of Ethernet LANs and engineering workstations. These file servers employed general-purpose operating systems on general-purpose hardware, usually the same machines sold as computing servers, or even workstations. This solution is still popular today, as file servers built with Microsoft and Linux operating systems employ standard server hardware and software.

Not long after general-purpose network file servers became popular in workgroups in the 1980s, Auspex of Santa Clara, Calif., developed an aggressively specialized NAS system using custom hardware, an unusual interconnect, asymmetric multiprocessing,

Figure 4. Function and network links in the case studies.

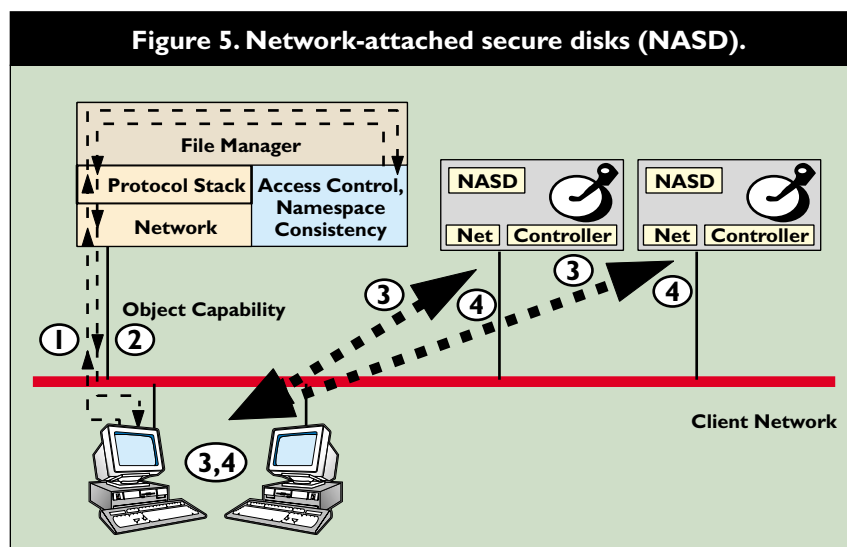


and a specialized operating and file system. Auspex file servers provided higher performance and an early example of the benefits of consolidating the storage resources of multiple workgroups.

Novell of Provo, Ut., and Network Appliance of Sunnyvale, Calif., chose a less-aggressive form of specialization, using general-purpose high-performance workstations and specialized (streamlined) operating and file systems [5]. Because of the performance improvement in the “killer micros” of the early 1990s, this approach was cost-effective while still providing high availability and simplified storage management through specialized software.

Today, there is a resurgence of storage hardware and software specialization in a variety of advanced development laboratories. One theory for this resurgence, beyond increasing market size, follows an analogy with network infrastructure. Some people reason that just as general-purpose routers gave way to specialized hardware and software switches now being scaled to terabit speeds, storage infrastructure should also be specialized around very high-bandwidth internal-interconnection networks. The major alternative to this “super fileservers” approach is the use of PC clusters with a low-latency cluster interconnect based on network interface cards offloading protocol processing from each machine’s main processor. Such network-specialized cluster approaches also require extensively specialized software; Network Appliance and Intel of Santa Clara, Calif., have proposed such a new file system architecture and protocol called the Direct Access File System [9].

**Less-specialized SAN systems.** For block servers



**Table 2. Case studies.**

	Storage Appliance	NASD	Petal	iSCSI
Interface Abstraction	files + directories	file objects	blocks	blocks
Geographic distribution			additional round trips	additional round trips
Trusted components	client OS	metadata server	client OS, network	client OS, network
Load-balance commands	no	yes (reads/writes)	yes	no
Consistent concurrent access	serialized	distributed locking	distributed locking	serialized
Redundancy for availability	inside the box	across the cluster	across the cluster	inside the box
Bottlenecks	controller	metadata changes	configuration changes	controller
Speed of deployment limits	save/restore files	online reallocation	online reallocation	stop, copy, reboot
Control and data	same path	separated paths	same path	same path
Resources consolidated	inside the box	by metadata server	server cluster	inside the box
Management executed by	controller	metadata server	any server	controller

with many disks, SCSI has been the leading storage-command protocol for almost 20 years. Over this time, SCSI command-level capabilities have evolved slowly, but the physical implementation has changed dramatically. As with most communication protocols, SCSI can be implemented as a layering of physical signal management, signaling and congestion control, and command interpretation. As technology advances drove changes in SCSI's lower levels, these levels have been replaced by faster, more flexible technologies. Fibre Channel is the dominant high-performance SAN technology today, using the same physical wires and transmission mechanisms as Gigabit Ethernet, although its signaling and congestion control is specialized to SCSI.

This trend toward generalization in SAN systems was also seen more than 15 years ago in Digital Equipment Corp.'s VAXclusters, which first used a proprietary physical interconnect (VAX-CI), then moved to

the more standard and more general-purpose Ethernet [6].

## Four Architectures

It may be too early to know how the changing levels of specialization in storage systems will affect how computers are built and sold. We can, however, identify the range of possible outcomes, particularly four architectures under active or recent research and development: storage appliances, iSCSI, NASD, and Petal. Storage appliances, commercially available today, are nonclustered specialized NAS systems. iSCSI is the least-specialized SAN interface. Petal and NASD were experimental research projects for clustering storage without server bottlenecks.

Figure 4 and Table 2 compare these architectures. The figure contrasts the organization of function into boxes and numbers of network transits when accessing data. (In it, boxes are computers; horizontal lines are communication paths; vertical lines are internal and external interfaces; LAN is an Internet network, like Ethernet; and SAN is a SCSI network like Fibre Channel.) The table shows each of the four case studies'

answers to the critical architectural questions discussed earlier.

**Storage appliances.** Storage appliances are NAS systems intended to be especially simple to manage, ranging from Network Appliance's terabyte servers to a disk drive with an Ethernet plug [5]. The Snap! server from Quantum of Milpitas, Calif., is an example of the lowest-price storage appliance available today. Externally, it is a standard NAS server. Internally, it includes a low-cost small-form-factor, single-board PC attached to one or two PC disk drives and packaged in a box smaller than the typical PC, with only a power cable and an Ethernet connector as external attachments.

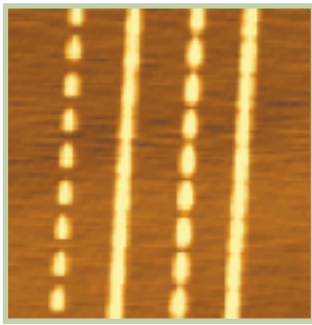
Snap! servers consist of two printed circuit boards: one for the computer, one for the disk-drive controller. A more specialized storage appliance might combine these boards, integrate their processors, and combine their code and datapaths. Cables, connec-

tors, boards, chips, and memory may be eliminated and made less costly. However, because storage devices are required to be one of the most reliable parts of any computer system, integrating NAS software into the disk-drive controller may initially decrease reliability and inhibit product acceptance. Moreover, the hardware design of today's disk controllers assumes the microprocessor on the disk does not need to read much of the data as it goes by, but NAS software typically moves every byte through the processor multiple times. Accordingly, hardware integration may be less than straightforward.

When the integration of NAS server and disk controller is achieved, disk-drive manufacturers will have a significant cost advantage over NAS system vendors offering two-board solutions. Thus, this highly inte-

Each client temporarily functions as a server for the files whose metadata it has in cache. Unfortunately, this role as temporary server increases the risk of security breaches, because devices continue trusting any client, and clients are notoriously easy to penetrate, especially by partially authorized employees.

A more secure asymmetric control-and-data architecture places the metadata at the device rather than at the client. NASD stored file descriptors and indirect blocks in the device permanently and let clients read and parse directories. NASD policy servers construct authorization tokens (capabilities) that clients digitally sign on every request and that NASD devices test on every command. By storing file metadata in the device, NASD offered a file interface abstraction. However, it did not offer directory operations,



**NEW NAS SYSTEMS CAN BE INSTALLED AND CONFIGURED WITHOUT INTERRUPTING THE EXECUTION OF CLIENT MACHINES. AND NAS USERS ON DIFFERENT MACHINES CAN SEE AND SHARE THE SAME COLLECTION OF FILES WITHOUT SPECIAL EFFORT.**

grated specialization will be driven by user requirements for file servers in the lowest-price packages.

Unfortunately, users want large-capacity configurations more often than they want small-capacity configurations. Hence, storage-appliance vendors concentrate on making each appliance as easy to configure as possible, then rack-mount many appliances together. Today, this combination of low cost and manual management of aggregation is popular for providing information content over the Web.

**NASD.** NASD was a research project at Carnegie Mellon University (beginning in 1995) pursuing the notion that aggregation of storage devices is best managed through a central policy server (possibly a cluster of servers), while most commands and data transfers move directly between device and client, bypassing the server; Figure 5 outlines this asymmetric control and datapath [3]. Upon approval by the central server, clients can access (directly and in parallel) all devices containing data of interest.

There are many ways to ensure that the server in such asymmetric systems controls client access to storage. The most common trusts client operating systems to request and cache file system metadata [4].

because it assumed data is spread over multiple NASDs and that the file system's directories are globally defined. Instead, a NASD policy server stores directory information in NASD files clients can parse without the device recognizing the difference between a regular file and a directory.

**Petal.** Petal was a research project at Compaq's Systems Research Center [7] based on arrays of storage-appliance-like disk "bricks" but offering a block-oriented interface rather than a file interface. Petal scaled by splitting the controller function over a cluster of controllers, any one of which had access to consistent global state. As a Petal system's capacity grew, so did the number of Petal servers in the cluster along with the performance they sustained. Logically, Petal could be viewed as a RAID system implemented on a symmetric multiprocessor, though it used distributed consensus algorithms instead of shared memory for global state.

For administrators, management in a Petal system was especially easy; any Petal server had access to global state, because servers could be added or lost at any time, and because device configurations could be changed at any time.



Petal was built using Internet (LAN) protocols but logically designed SAN interface. The availability of iSCSI, discussed next, provided an ideal alternative to Petal's custom LAN-based SAN protocol.

**iSCSI.** Layering a block-level SAN protocol over Internet protocols, such as was demonstrated by the Netstation project at the University of Southern California's Information Sciences Institute starting in 1991, is a natural way for storage to exploit the influence of the Internet [12]. Beginning earlier this year, an Internet Engineering Task Force (IETF) working group has sought to standardize a block-level command-and-data-movement system over the principle Internet protocol, conveniently named the Internet Protocol (IP). Known informally as iSCSI, for Internet SCSI, the IPS (IP Storage) working group is also chartered to work on security, naming, discovery, configuration, and quality of service for IP storage.

A number of startup companies have declared their interest in implementing iSCSI in storage "routers" that may turn out to be similar to RAID systems. Like Fibre Channel's command protocol FCP, iSCSI is a SAN interface. Although it does not provide file system functionality, it should interoperate with systems designed for other SANs, once the storage industry figures out how to configure and tune different SAN interconnects interoperably.

iSCSI is a step toward generalization in storage-device networking. Taking advantage of the existing body of work on Internet communication protocols and media, it intends to give storage networking the scaling properties of IP networks. Not yet clear to IPS working group observers, however, is how much of the Internet protocol suite will be used without modification. For example, most Internet applications employ a transport mechanism called TCP to reliably deliver data over IP. The IETF working group uses TCP as it is defined today; other groups are proposing small changes to TCP to make high-speed data transfer more efficient; still others are proposing an entirely different congestion-control algorithm they think is better for storage traffic.

## Conclusion

Storage systems are becoming the dominant investment in corporate data centers and a crucial asset in e-commerce, making the rate of growth of storage a strategic business problem and a major business opportunity for storage vendors. In order to satisfy user needs, storage systems should consolidate resources, deploy quickly, be centrally managed, be highly available, and allow data sharing. It should also be possible to distribute them over global distances, make them secure against external and inter-

nal abuse, and scale their performance with capacity. Putting storage in specialized systems and accessing it from clients across a network provides significant advantages for users. Moreover, the most apparent difference between the NAS and SAN versions of network storage—use of Ethernet in NAS and Fibre Channel in SAN—is not a core difference and may soon not even be a recognizable difference. Instead, we may have NAS servers that look like disks, disks that connect to and operate on Ethernet, arrays of disk bricks that, as far as the user is concerned, function as one big disk, and arrays of smart disks that verify every command against the rights of individual users. **■**

## REFERENCES

1. Benner, A. *Fibre Channel: Gigabit Communications and I/O for Computer Networks*. McGraw Hill, New York, 1996.
2. Callaghan, B. *NFS Illustrated*. Addison Wesley Publishing Co., Reading, Mass., 2000.
3. Gibson, G., et al. A cost-effective, high-bandwidth storage architecture. In *Proceedings of the ACM 8th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)* (San Jose, Calif., Oct). ACM Press, New York, 1998, 92–103; see also [www.pdl.cs.cmu.edu](http://www.pdl.cs.cmu.edu).
4. Hartman, J. and Ousterhout, J. The Zebra striped network file system. In *Proceedings of ACM Symposium on Operating Systems Principles (SOSP)* (Asheville, N.C., Dec.). ACM Press, New York, 1993, 29–43.
5. Hitz, D., Lau, J., and Malcolm, M. File systems design for an NFS file server appliance. In *USENIX Winter 1994 Technical Conference Proceedings* (San Francisco, Jan. 1994).
6. Kronenberg, N., et al. VAXclusters: A closely coupled distributed system. *ACM Transact. Comput. Syst. (TOCS)* 4, 2 (May 1986), 130–146.
7. Lee, E. and Thekkath, C. Petal: Distributed virtual disks. In *Proceedings of the ACM 7th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)* (Cambridge, Mass., Oct). ACM Press, New York, 1996, 84–92.
8. McKusick, M., et al. A fast file system for Unix. *ACM Transact. Comput. Syst. (TOCS)* 2, 3 (Aug. 1984).
9. Network Appliance, Inc. *DAFS: Direct Access File System Protocol, Version 0.53* (July 31, 2000); see [www.dafscollaborative.org](http://www.dafscollaborative.org).
10. Sachs, M., Leff, A., and Sevigny, D. LAN and I/O convergence: A survey of the issues. *IEEE Comput.* 27, 12 (Dec. 1994), 24–32.
11. Satran, J., et al. *iSCSI (Internet SCSI), IETF draft-satran-isci-01.txt* (Jul. 10, 2000); see [www.ece.cmu.edu/~ips](http://www.ece.cmu.edu/~ips).
12. Van Meter, R., Finn, G., and Hotz, S. VISA: Netstation's virtual Internet SCSI adapter. In *Proceedings of the ACM 8th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)* (San Jose, Calif., Oct.). ACM Press, New York, 1998, 71–80; see also [www.isi.edu/netstation/](http://www.isi.edu/netstation/).

**GARTH A. GIBSON** ([garth@panasas.com](mailto:garth@panasas.com), [garth@cs.cmu.edu](mailto:garth@cs.cmu.edu)) is the chief technology officer of Panasas, Inc., Pittsburgh, PA, and an associate professor (on leave) in the School of Computer Science at Carnegie Mellon University, Pittsburgh, PA.

**RODNEY VAN METER** ([rdv@alumni.caltech.edu](mailto:rdv@alumni.caltech.edu)) is a senior software engineer in Nokia Internet Communications in Santa Cruz, CA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.