Chapter 7
Network-Attached Storage

Network-attached storage (NAS) is an IP-based file-sharing device attached to a local area network. NAS provides the advantages of server consolidation by eliminating the need for multiple file servers. It provides storage consolidation through file-level data access and sharing. NAS is a preferred storage solution that enables clients to share files quickly and directly with minimum storage management overhead. NAS also helps to eliminate bottlenecks that users face when accessing files from a general-purpose server.

KEY CONCEPTS

- NAS Device
- Remote File Sharing
- NAS Connectivity and Protocols
- NAS Performance and Availability
- MTU and Jumbo Frames

NAS uses network and file-sharing protocols to perform filing and storage functions. These protocols include TCP/IP for data transfer and CIFS and NFS for remote file service. NAS enables both UNIX and Microsoft Windows users to share the same data seamlessly. To enable data sharing, NAS typically uses NFS for UNIX, CIFS for Windows, and File Transfer Protocol (FTP) and other protocols for both environments. Recent advancements in networking technology have enabled NAS to scale up to enterprise requirements for improved performance and reliability in accessing data.

A NAS device is a dedicated, high-performance, high-speed, single-purpose file serving and storage system. NAS serves a mix of clients and servers over an IP network. Most NAS devices support multiple interfaces and networks.

A NAS device uses its own operating system and integrated hardware, software components to meet specific file service needs. Its operating system is optimized for file I/O and, therefore, performs file I/O better than a general-purpose server. As a result, a NAS device can serve more clients than traditional file servers, providing the benefit of server consolidation.

This chapter describes the components of NAS, different types of NAS implementations, the file-sharing protocols, and the transport and network layer protocols used in NAS implementations. The chapter also explains NAS design considerations and factors that affect NAS performance.

7.1 General-Purpose Servers vs. NAS Devices

A NAS device is optimized for file-serving functions such as storing, retrieving, and accessing files for applications and clients. As shown in Figure 7-1, a general-purpose server can be used to host any application, as it runs a generic operating system. Unlike a general-purpose server, a NAS device is dedicated to file-serving. It has a real-time operating system dedicated to file serving by using open-standard protocols. Some NAS vendors support features such as native clustering for high availability.

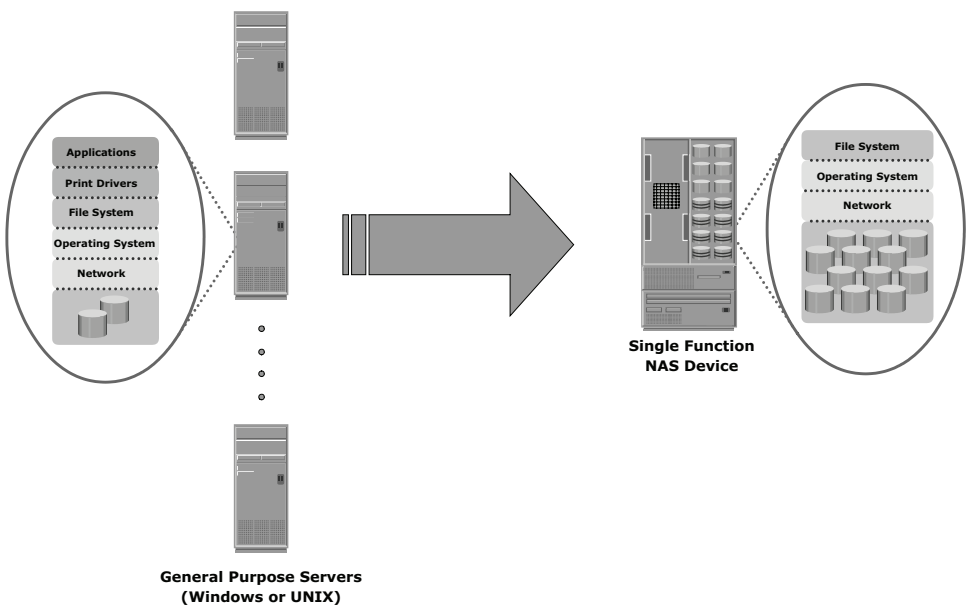

Figure 7-1: General purpose server vs. NAS device

7.2 Benefits of NAS

NAS offers the following benefits:

- **Supports comprehensive access to information:** Enables efficient file sharing and supports many-to-one and one-to-many configurations. The

many-to-one configuration enables a NAS device to serve many clients simultaneously. The one-to-many configuration enables one client to connect with many NAS devices simultaneously.

- **Improved efficiency:** Eliminates bottlenecks that occur during file access from a general-purpose file server because NAS uses an operating system specialized for file serving. It improves the utilization of general-purpose servers by relieving them of file-server operations.
- **Improved flexibility:** Compatible for clients on both UNIX and Windows platforms using industry-standard protocols. NAS is flexible and can serve requests from different types of clients from the same source.
- **Centralized storage:** Centralizes data storage to minimize data duplication on client workstations, simplify data management, and ensures greater data protection.
- **Simplified management:** Provides a centralized console that makes it possible to manage file systems efficiently.
- **Scalability:** Scales well in accordance with different utilization profiles and types of business applications because of the high performance and low-latency design.
- **High availability:** Offers efficient replication and recovery options, enabling high data availability. NAS uses redundant networking components that provide maximum connectivity options. A NAS device can use clustering technology for failover.
- **Security:** Ensures security, user authentication, and file locking in conjunction with industry-standard security schemas.

7.3 NAS File I/O

NAS uses file-level access for all of its I/O operations. File I/O is a high-level request that specifies the file to be accessed, but does not specify its logical block address. For example, a file I/O request from a client may specify reading 256 bytes from byte number 1152 onward in a specific file. Unlike block I/O, there is no disk volume or disk sector information in a file I/O request. The NAS operating system keeps track of the location of files on the disk volume and converts client file I/O into block-level I/O to retrieve data.

The NAS operating system issues a block I/O request to fulfill the file read and write requests that it receives. The retrieved data is again converted to file-level I/O for applications and clients.

7.3.1 File Systems and Remote File Sharing

A file system is a structured way of storing and organizing data files. Many file systems maintain a file access table to simplify the process of finding and accessing files.

7.3.2 Accessing a File System

A file system must be mounted before it can be used. In most cases, the operating system mounts a local file system during the boot process. The mount process creates a link between the file system and the operating system. When mounting a file system, the operating system organizes files and directories in a tree-like structure and grants the user the privilege of accessing this structure. The tree is rooted at a mount point that is named using operating system conventions. Users and applications can traverse the entire tree from the root to the leaf nodes. Files are located at leaf nodes, and directories and subdirectories are located at intermediate roots. The relationship between the user and the file system terminates when the file system is unmounted. Figure 7-2 shows an example of the UNIX directory structure under UNIX operating environments.

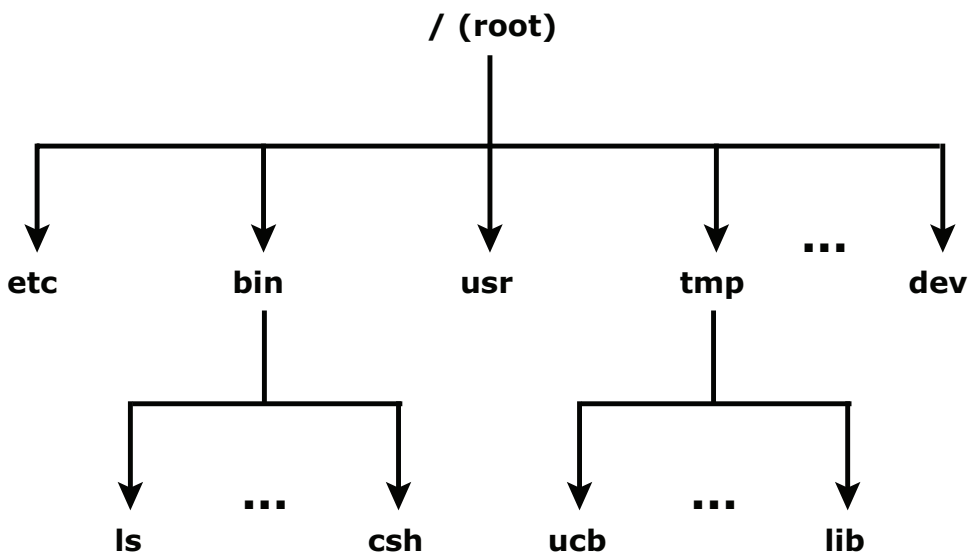

Figure 7-2: UNIX directory structure

7.3.3 File Sharing

File sharing refers to storing and accessing data files over a network. In a file-sharing environment, a user who creates the file (the creator or owner of a file)

determines the type of access to be given to other users (read, write, execute, append, delete, and list) and controls changes to the file. When multiple users try to access a shared file at the same time, a protection scheme is required to maintain data integrity and, at the same time, make this sharing possible.

File Transfer Protocol (FTP), distributed file systems, and a client/server model that uses a file-sharing protocol are some examples of implementations of file-sharing environments.

FTP is a client/server protocol that enables data transfer over a network. An FTP server and an FTP client communicate with each other using TCP as the transport protocol. FTP, as defined by the standard, is not a secure method of data transfer because it uses unencrypted data transfer over a network. FTP over Secure Shell (SSH) adds security to the original FTP specification.

A *distributed file system (DFS)* is a file system that is distributed across several hosts. A DFS can provide hosts with direct access to the entire file system, while ensuring efficient management and data security.

The traditional *client/server model*, which is implemented with file-sharing protocols, is another mechanism for remote file sharing. In this model, the clients mount remote file systems that are available on dedicated file servers. The standard client/server file-sharing protocols are NFS for UNIX and CIFS for Windows. NFS and CIFS enable the owner of a file to set the required type of access, such as read-only or read-write, for a particular user or group of users.

In both of these implementations, users are unaware of the location of the file system. In addition, a *name service*, such as Domain Name System (DNS), Lightweight Directory Access Protocol (LDAP), and Network Information Services (NIS), helps users identify and access a unique resource over the network. A *naming service protocol* creates a namespace, which holds the unique name of every network resource and helps recognize resources on the network.

7.4 Components of NAS

A NAS device has the following components (see Figure 7-3):

- NAS head (CPU and Memory)
- One or more network interface cards (NICs), which provide connectivity to the network. Examples of NICs include Gigabit Ethernet, Fast Ethernet, ATM, and Fiber Distributed Data Interface (FDDI).
- An optimized operating system for managing NAS functionality
- NFS and CIFS protocols for file sharing
- Industry-standard storage protocols to connect and manage physical disk resources, such as ATA, SCSI, or FC

The NAS environment includes clients accessing a NAS device over an IP network using standard protocols.

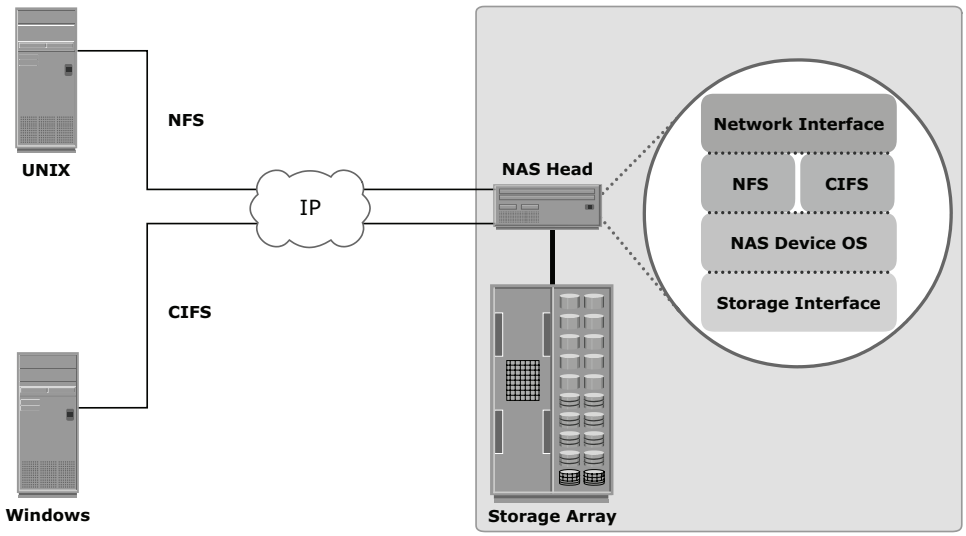

Figure 7-3: Components of NAS

7.5 NAS Implementations

As mentioned earlier, there are two types of NAS implementations: integrated and gateway. The *integrated NAS* device has all of its components and storage system in a single enclosure. In *gateway* implementation, NAS head shares its storage with SAN environment.

7.5.1 Integrated NAS

An integrated NAS device has all the components of NAS, such as the NAS head and storage, in a single enclosure, or frame. This makes the integrated NAS a self-contained environment. The NAS head connects to the IP network to provide connectivity to the clients and service the file I/O requests. The storage consists of a number of disks that can range from low-cost ATA to high-throughput FC disk drives. Management software manages the NAS head and storage configurations.

An integrated NAS solution ranges from a low-end device, which is a single enclosure, to a high-end solution that can have an externally connected storage array.

A low-end appliance-type NAS solution is suitable for applications that a small department may use, where the primary need is consolidation of storage, rather than high performance or advanced features such as disaster recovery and business continuity. This solution is fixed in capacity and might not be upgradable beyond its original configuration. To expand the capacity, the solution must be scaled by deploying additional units, a task that increases management overhead because multiple devices have to be administered.

In a high-end NAS solution, external and dedicated storage can be used. This enables independent scaling of the capacity in terms of NAS heads or storage. However, there is a limit to scalability of this solution.

7.5.2 Gateway NAS

A gateway NAS device consists of an independent NAS head and one or more storage arrays. The NAS head performs the same functions that it does in the integrated solution; while the storage is shared with other applications that require block-level I/O. Management functions in this type of solution are more complex than those in an integrated environment because there are separate administrative tasks for the NAS head and the storage. In addition to the components that are explicitly tied to the NAS solution, a gateway solution can also utilize the FC infrastructure, such as switches, directors, or direct-attached storage arrays.

The gateway NAS is the most scalable because NAS heads and storage arrays can be independently scaled up when required. Adding processing capacity to the NAS gateway is an example of scaling. When the storage limit is reached, it can scale up, adding capacity on the SAN independently of the NAS head. Administrators can increase performance and I/O processing capabilities for their environments without purchasing additional interconnect devices and storage. Gateway NAS enables high utilization of storage capacity by sharing it with SAN environment.

7.5.3 Integrated NAS Connectivity

An integrated solution is self-contained and can connect into a standard IP network. Although the specifics of how devices are connected within a NAS implementation vary by vendor and model. In some cases, storage is embedded within a NAS device and is connected to the NAS head through internal connections, such as ATA or SCSI controllers. In others, the storage may be external but connected by using SCSI controllers. In a high-end integrated NAS model, external storage can be directly connected by FC HBAs or by dedicated FC switches. In the case of a low-end integrated NAS model, backup traffic is shared on the same public IP network along with the regular client access traffic. In the case of a high-end integrated NAS model, an isolated backup network

can be used to segment the traffic from impeding client access. More complex solutions may include an intelligent storage subsystem, enabling faster backup and larger capacities while simultaneously enhancing performance. Figure 7-4 illustrates an example of integrated NAS connectivity.

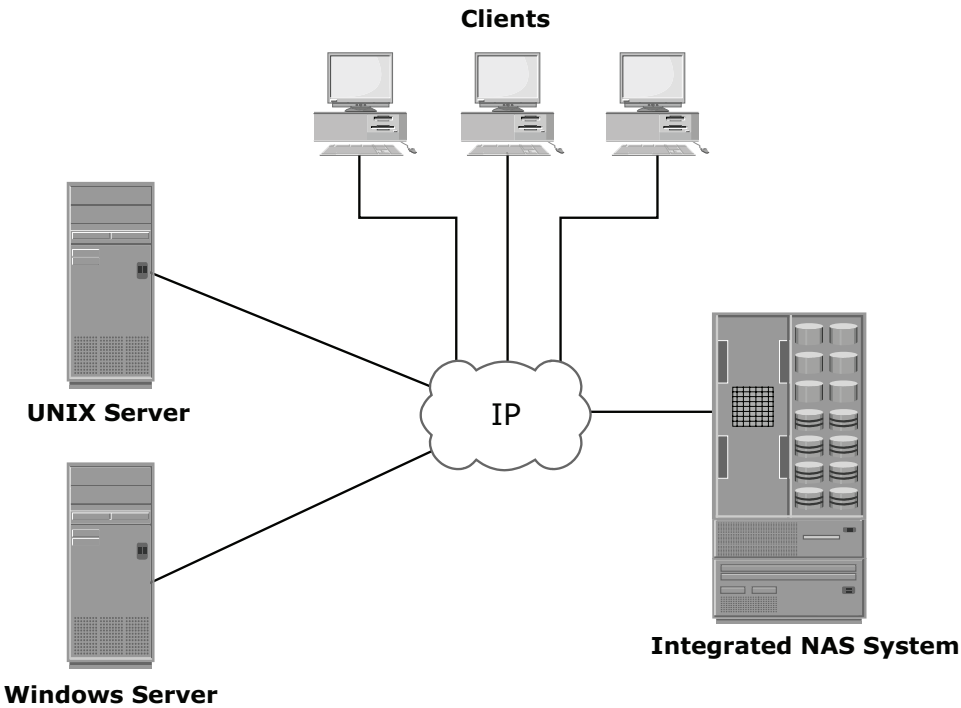

Figure 7-4: Integrated NAS connectivity

7.5.4 Gateway NAS Connectivity

In a gateway solution, front-end connectivity is similar to that in an integrated solution. An integrated environment has a fixed number of NAS heads, making it relatively easy to determine IP networking requirements. In contrast, networking requirements in a gateway environment are complex to determine due to scalability options. Adding more NAS heads may require additional networking connectivity and bandwidth.

Communication between the NAS gateway and the storage system in a gateway solution is achieved through a traditional FC SAN. To deploy a stable NAS solution, factors such as multiple paths for data, redundant fabrics, and load distribution must be considered. Figure 7-5 illustrates an example of gateway NAS connectivity.

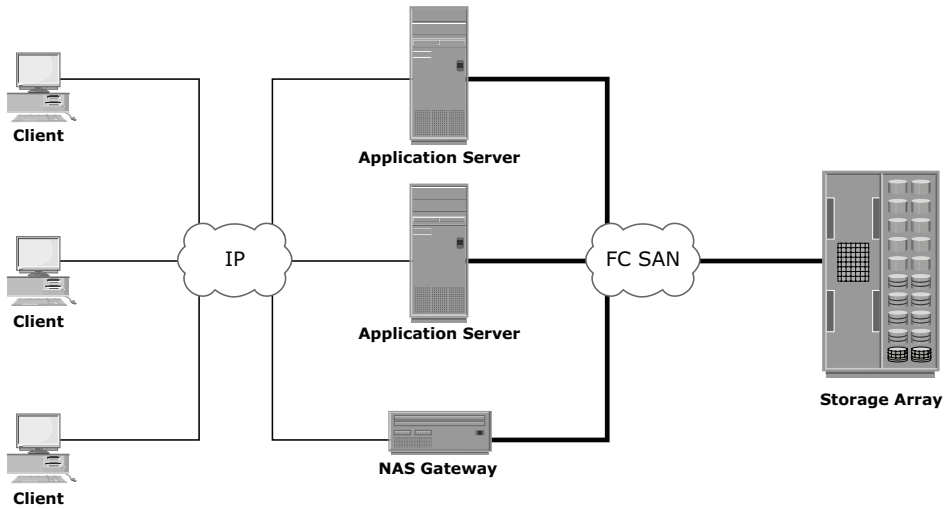

Figure 7-5: Gateway NAS connectivity

Implementation of a NAS gateway solution requires analysis of current SAN environment. This analysis is required to determine the feasibility of introducing a NAS workload to the existing SAN. Analyze the SAN to determine whether the workload is primarily read or write, or random or sequential. Determine the predominant I/O size in use. In general, sequential workloads have large I/Os. Typically, NAS workloads are random with small I/O size. Introducing sequential workload with random workloads can be disruptive to the sequential workload. Therefore, it is recommended to separate the NAS and SAN disks. Also, determine whether the NAS workload performs adequately with the configured cache in the storage subsystem.

7.6 NAS File-Sharing Protocols

Most NAS devices support multiple file service protocols to handle file I/O requests to a remote file system. As mentioned earlier, NFS and CIFS are the common protocols for file sharing. NFS is predominantly used in UNIX-based operating environments; CIFS is used in Microsoft Windows–based operating environments.

These file sharing protocols enable users to share file data across different operating environments and provide a means for users to migrate transparently from one operating system to another.

7.6.1 NFS

NFS is a client/server protocol for file sharing that is most commonly used on UNIX systems. NFS was originally based on the connectionless *User Datagram Protocol* (UDP). It uses a machine-independent model to represent user data. It also uses Remote Procedure Call (RPC) as a method of interprocess communication between two computers. The NFS protocol provides a set of RPCs to access a remote file system for the following operations:

- Searching files and directories
- Opening, reading, writing to, and closing a file
- Changing file attributes
- Modifying file links and directories

NFS uses the mount protocol to create a connection between the client and the remote system to transfer data. NFS (NFSv3 and earlier) is a *stateless* protocol, which means that it does not maintain any kind of table to store information about open files and associated pointers. Therefore, each call provides a full set of arguments to access files on the server. These arguments include a file name and a location, a particular position to read or write, and the versions of NFS.

Currently, three versions of NFS are in use:

- **NFS version 2 (NFSv2):** Uses UDP to provide a stateless network connection between a client and a server. Features such as locking are handled outside the protocol.
- **NFS version 3 (NFSv3):** The most commonly used version, it uses UDP or TCP, and is based on the stateless protocol design. It includes some new features, such as a 64-bit file size, asynchronous writes, and additional file attributes to reduce re-fetching.
- **NFS version 4 (NFSv4):** This version uses TCP and is based on a stateful protocol design. It offers enhanced security.

7.6.2 CIFS

CIFS is a client/server application protocol that enables client programs to make requests for files and services on remote computers over TCP/IP. It is a public, or open, variation of Server Message Block (SMB) protocol.

The CIFS protocol enables remote clients to gain access to files that are on a server. CIFS enables file sharing with other clients by using special locks.

File names in CIFS are encoded using unicode characters. CIFS provides the following features to ensure data integrity:

- It uses file and record locking to prevent users from overwriting the work of another user on a file or a record.
- It runs over TCP.
- It supports fault tolerance and can automatically restore connections and reopen files that were open prior to interruption. The fault tolerance features of CIFS depend on whether an application is written to take advantage of these features. Moreover, CIFS is a stateful protocol because the CIFS server maintains connection information regarding every connected client. In the event of a network failure or CIFS server failure, the client receives a disconnection notification. User disruption is minimized if the application has the embedded intelligence to restore the connection. However, if the embedded intelligence is missing, the user has to take steps to reestablish the CIFS connection.

  Users refer to remote file systems with an easy-to-use file naming scheme:

  ```
  \\server\share or \\servername.domain.suffix\share.
  ```

7.7 NAS I/O Operations

The NFS and CIFS protocols handle file I/O requests to a remote file system, which is managed by the NAS device. The process of NAS I/O is as follows:

1. The requestor packages an I/O request into TCP/IP and forwards it through the network stack. The NAS device receives this request from the network.
2. The NAS device converts the I/O request into an appropriate physical storage request, which is a block-level I/O, and then performs the operation against the physical storage pool.
3. When the data is returned from the physical storage pool, the NAS device processes and repackages the data into an appropriate file protocol response.
4. The NAS device packages this response into TCP/IP again and forwards it to the client through the network.

Figure 7-6 illustrates this process.

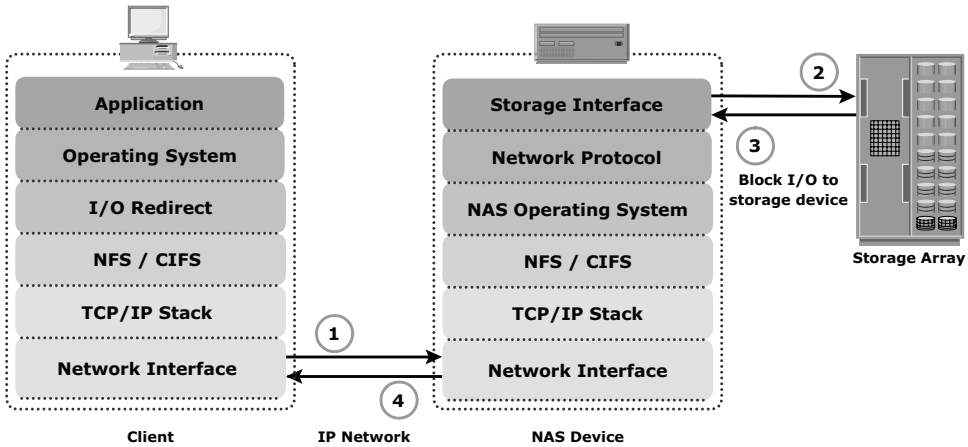

Figure 7-6: NAS I/O operation

7.7.1 Hosting and Accessing Files on NAS

Following are the steps required to host files and permit users to access the hosted files on a NAS device:

1. **Create storage array volumes:** Create volumes on the storage array and assign Logical Unit Numbers (LUN) to the volumes. Present the newly created volumes to the NAS device.
2. **Create NAS Volumes:** Perform a discovery operation on the NAS device, to recognize the new array-volumes and create NAS Volumes (logical volumes). Multiple volumes from the storage array may be combined to form large NAS volumes.
3. **Create NAS file systems:** Create NAS file systems on the NAS volumes.
4. **Mount file systems:** Mount the created NAS file system on the NAS device.
5. **Access the file systems:** Publish the mounted file systems on the network using NFS or CIFS for client access.

7.8 Factors Affecting NAS Performance and Availability

As NAS uses IP network, bandwidth and latency issues associated with IP affect NAS performance. Network congestion is one of the most significant sources

of latency (Figure 7-7) in a NAS environment. Other factors that affect NAS performance at different levels are:

1. **Number of hops:** A large number of hops can increase latency because IP processing is required at each hop, adding to the delay caused at the router.
2. **Authentication with a directory service such as LDAP, Active Directory, or NIS:** The authentication service must be available on the network, with adequate bandwidth, and must have enough resources to accommodate the authentication load. Otherwise, a large number of authentication requests are presented to the servers, increasing latency. Authentication adds to latency only when authentication occurs.
3. **Retransmission:** Link errors, buffer overflows, and flow control mechanisms can result in retransmission. This causes packets that have not reached the specified destination to be resent. Care must be taken when configuring parameters for speed and duplex settings on the network devices and the NAS heads so that they match. Improper configuration may result in errors and retransmission, adding to latency.
4. **Overutilized routers and switches:** The amount of time that an overutilized device in a network takes to respond is always more than the response time of an optimally utilized or underutilized device. Network administrators can view vendor-specific statistics to determine the utilization of switches and routers in a network. Additional devices should be added if the current devices are overutilized.
5. **File/directory lookup and metadata requests:** NAS clients access files on NAS devices. The processing required before reaching the appropriate file or directory can cause delays. Sometimes a delay is caused by deep directory structures and can be resolved by flattening the directory structure. Poor file system layout and an overutilized disk system can also degrade performance.
6. **Overutilized NAS devices:** Clients accessing multiple files can cause high utilization levels on a NAS device which can be determined by viewing utilization statistics. High utilization levels can be caused by a poor file system structure or insufficient resources in a storage subsystem.
7. **Overutilized clients:** The client accessing CIFS or NFS data may also be overutilized. An overutilized client requires longer time to process the responses received from the server, increasing latency. Specific performance-monitoring tools are available for various operating systems to help determine the utilization of client resources.

Configuring VLANs and setting proper Maximum Transmission Unit (MTU) and TCP window size can improve NAS performance. Link aggregation and redundant network configurations ensure high availability.

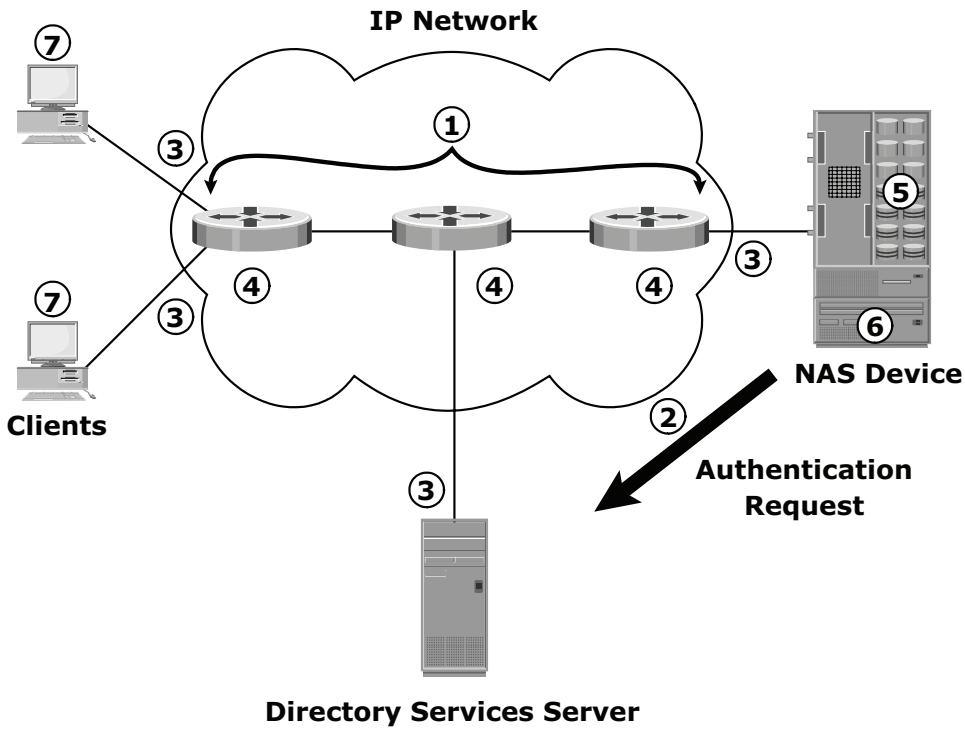

Figure 7-7: Causes of latency

A *virtual LAN (VLAN)* is a switched network that is logically segmented by functions, project teams, or applications, regardless of the user's physical location. A VLAN is similar to a physical LAN except that the VLAN enables the grouping of end stations even if they are not physically located on the same network segment. VLAN is a layer 2 (data link layer) construct. A network switch can be divided among multiple VLANs, enabling better utilization of port density and reducing the overall cost of deploying a network infrastructure.

A VLAN can control the overall broadcast traffic. The broadcast traffic on one VLAN is not transmitted outside that VLAN, which substantially reduces broadcast overhead, makes bandwidth available for applications, and reduces the network's vulnerability to broadcast storms.

VLANs are also used to provide security firewalls, restrict individual user access, flag network intrusions, and control the size and composition of the broadcast domain.

The *MTU* setting determines the size of the largest packet that can be transmitted without data fragmentation. *Path maximum transmission unit discovery* is the process of discovering the maximum size of a packet that can be sent across a network without fragmentation. The default MTU settings are specific for each protocol and depend on the type of NIC installed. The default MTU setting for an Ethernet interface card is 1,500 bytes. A feature called *jumbo frames* is used to send, receive, or transport Ethernet frames with an MTU of more than 1,500 bytes. The most common deployments of jumbo frames have an MTU of 9,000 bytes. Servers send and receive larger frames more efficiently than smaller ones in heavy network traffic conditions. Jumbo frames ensure increased efficiency because it takes fewer, larger frames to transfer the same amount of data, just as with existing Ethernet packets. Larger packets also reduce the amount of raw network bandwidth being consumed for the same amount of payload. Larger frames also help smooth the sudden I/O bursts.

The *TCP window size* is the maximum amount of data that can be on the network at any time for a connection. For example, if a pair of hosts is talking over a TCP connection that has a TCP window size of 64 KB, the sender can send only 64 KB of data and must then wait for an acknowledgment from the receiver. If the receiver acknowledges that all the data has been received, then the sender is free to send another 64 KB of data. If the sender receives an acknowledgment from the receiver that only the first 32 KB of data has been received, which can happen only if another 32 KB of data is in transit or was lost, the sender can only send another 32 KB of data because the transmission cannot have more than 64 KB of unacknowledged data outstanding.

In theory, the TCP window size should be set to the product of the available bandwidth of the network and the round-trip time of data sent over the network. For example, if a network has a bandwidth of 100 Mbps and the round-trip time is 5 milliseconds, the TCP window should be as follows:

100 Mb/s × .005 seconds = 524,288 bits

524,288 bits / 8 bits/byte = 65,536 bytes

The size of TCP window field that controls the flow of data is between 2 bytes and 65,535 bytes.

Link aggregation is the process of combining two or more network interfaces into a logical network interface, enabling higher throughput, load sharing or load balancing, transparent path failover, and scalability. Link aggregation in a NAS device combines channels to achieve redundancy of network connectivity. Due to link aggregation, multiple active Ethernet connections to the same switch appear as one link. If a connection or a port in the aggregation is lost, then all the network traffic on that link is redistributed across the remaining active connections. The primary purpose of the aggregation is high availability.

7.9 Concepts in Practice: EMC Celerra

EMC offers NAS solutions with the Celerra family of products. Celerra is available in either a gateway or an integrated configuration. EMC Celerra provides a dedicated, high-performance, high-speed communication infrastructure for file level I/Os. It uses a significantly streamlined or tuned operating system. It supports Network Data Management Protocol (NDMP) for backup, CIFS, NFS, FTP, and iSCSI. Celerra supports Fast Ethernet Channel by using PAgP or IEEE 802.3ad LACP to combine two or more data channels into one data channel for high availability. Visit `http://education.EMC.com/ismbook` for the latest information.

7.9.1 Architecture

Celerra consists of a Control Station and NAS heads (Data Movers). The *Data Mover* is a network and storage interface device and the *control station* is a management interface device.

Data Mover

The Data Mover is an independent, autonomous file server that transfers requested files to clients. Figure 7-8 shows the NS40 and NSX blade Data Movers. *Data Access in Real Time (DART)* is Celerra's specialized operating system, which runs on the Data Mover. This operating system is optimized for performing file operations to move data from the storage array to the network. DART supports standard network file access protocols, such as NFS, CIFS, and FTP.

Celerra can be configured as a network fail-safe device; DART minimizes disruption to data access caused by network failures. A logical device is created using physical ports or logical ports combined together to create redundant groups of ports. The logically grouped Data Mover network ports monitor network traffic on the ports. Celerra also contains an active fail-safe device port, which can sense traffic disruption. The standby or non-active port assumes the IP address and the MAC address, minimizing disruption to data access. Features of the Celerra Data Mover include the following:

- Dual Intel processors
- PCI or PCI-X support
- High-memory capacity
- Multiport network cards
- Fibre Channel connectivity to storage arrays
- A highly specialized operating system (DART)

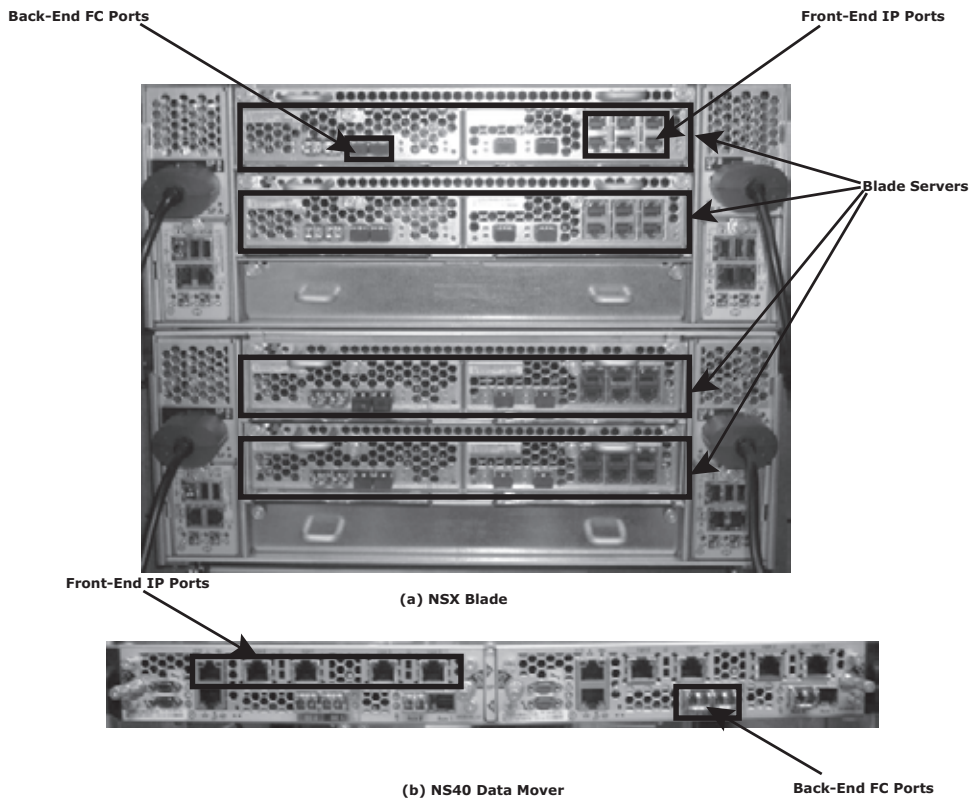

Figure 7-8: Celerra Data Mover

Creating and accessing a file system are the two important *client access functions* of the Celerra Data Mover. Creation of file systems is accomplished either manually or by using Automatic Volume Management (AVM). Accessing file systems is accomplished by performing an export in the UNIX environment or by publishing a share in the Microsoft Windows environment.

The Data Mover enables creation of multiple *virtual Data Movers* on a single physical Data Mover. A virtual Data Mover creates multiple virtual CIFS servers on each virtual Data Mover. The virtual Data Mover consolidates the file-serving functionality of multiple servers onto a Data Mover because each virtual Data Mover can maintain isolated CIFS servers with their own root file system environments. The virtual Data Mover also enables entire virtual environments to be loaded, unloaded, or even replicated between physical Data Movers.

The Celerra Data Mover offers two configuration options for NAS environments: a primary Data Mover or a standby Data Mover. A standby Data Mover

can be configured as a primary Data Mover for high availability configurations. A standby Data Mover is also provided for a group of primary Data Movers. Celerra Data Movers operate in one of the three modes, which affect the process of failover. In *Automatic mode*, the failover process occurs without first trying recovery; in *Retry mode*, the Data Mover is rebooted before failover; and in *Manual mode*, the failover is done manually.

Control Station

The *control station* provides dedicated processing capabilities to control, manage, and configure a NAS solution. The control station hosts the Linux operating system that is used to install, manage, and configure Data Movers and monitor the environmental conditions and performance of all components. The control station also provides high-availability features such as fault monitoring, fault recovery, fault reporting, call home, and remote diagnostics. Administrative functions are also accessible through the local console, SSH, or a Web browser.

Storage Connectivity

Celerra data movers connect to storage in two ways: integrated and gateway. In the integrated configuration, dedicated storage is assigned to Celerra (see Figure 7-9[a]). In this configuration, the control station is connected to the Data Movers via a private, internal IP network. Each Data Mover is directly connected to the storage array through dual Fibre Channel connections. The Data Movers provide an interface for the control station through both an internal Ethernet and a serial connection.

After DART is loaded, the Data Movers can be connected to the client network. The Data Movers are connected to the control station for remote management. The physical disks on the storage array are then partitioned through commands from the control station to create system volumes and data volumes for client access.

In a gateway configuration, Celerra is assigned separately provisioned storage within a shared storage array. Any capacity remaining within the array can be assigned to the conventional SAN hosts that are connected to the Fibre Switch (see Figure 7-9[b]) after appropriate zoning and LUN masking is performed. Each Data Mover is dual-connected to the storage array for redundancy through one or more Fibre Channel switches in the gateway configuration.

Client access and configuration steps are similar to that of an integrated connect.

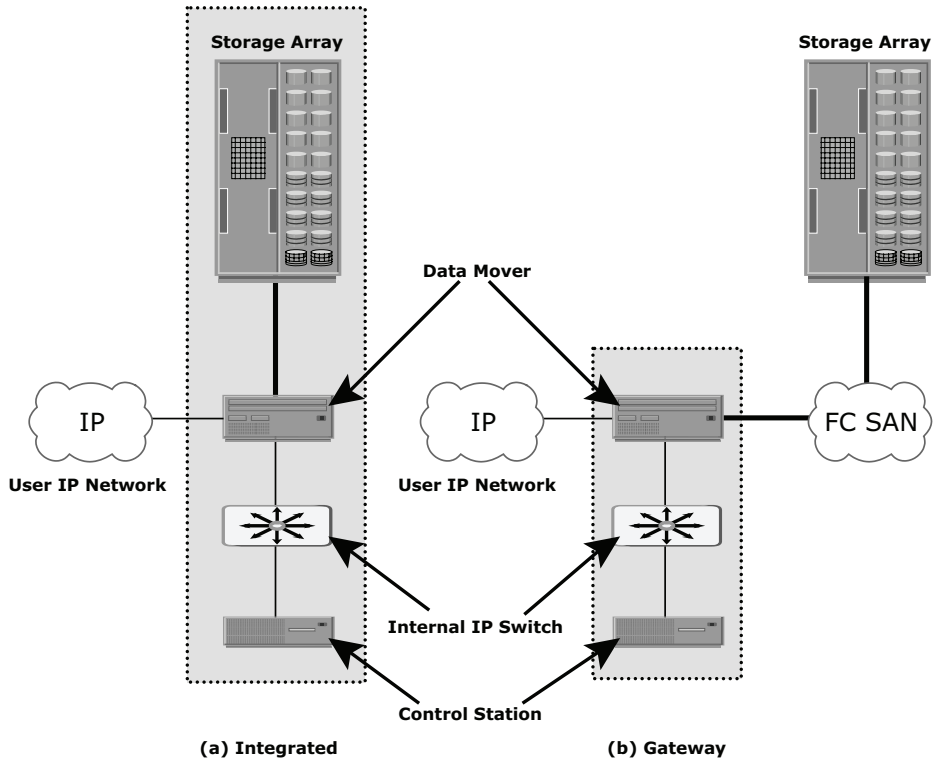

Figure 7-9: Celerra integrated and gateway connect

7.9.2 Celerra Product Family

This section describes some of the integrated and gateway NAS products available from EMC.

NS Series: Integrated

The back-end storage is directly connected to Celerra in the NS Series. The following models are offered:

- **NS20:** The NS20 comes in either an NS20 or an NS20FC configuration, and can have one or two X-Blade (Data Movers), and a single control station.
- **NS40:** This has a single or dual X-Blade configuration. This series provides high performance and capacity.

NS Series: Gateway

Gateway Data Movers share the back-end storage with the SAN hosts. The following models are offered:

- **NS40G:** This is an entry-level gateway, which delivers high performance, provides high availability, and requires simple management.
- **NSX**: The NSX system is the most highly redundant model in the Celerra family of products. It supports 4–8 X-Blades that come in an X-Blade Enclosure, and provides redundant management switches in each enclosure and dual control stations.

7.9.3 Celerra Management Software

EMC Celerra can be managed through either a command-line interface or the Celerra Manager GUI.

- **CLI Management:** This can be accessed from the control station through the SSH (secure shell) interface tool, or PUTTY. CLI can be used for scripting common repetitive tasks that may run on a predetermined schedule to ease administrative burdens. It has approximately 80 UNIX-like commands.
- **GUI Management—Celerra Manager:** GUI management through the Celerra Manager has the following two options—*Celerra Manager Basic Edition,* or *Celerra Manager Advanced Edition,* which is licensed separately and provides advanced features.

Summary

Decisions regarding storage infrastructure are based on maintaining the balance between cost and performance. Organizations look for the performance and scalability of SAN combined with the ease of use and lower total cost of ownership of NAS solutions. The convergence of SAN and NAS is inevitable because of advances in storage networking technology. Both SAN and NAS have enjoyed unique advantages in enterprises, and the advances in IP technology have scaled NAS solutions to meet the demands of performance-sensitive applications.

Cost and ease of use drive the application's needs in terms of performance and capacity. Although NAS invariably imposes higher protocol overhead, NAS applications tend to be most efficient for file-sharing tasks, such as NFS in UNIX, and CIFS in Microsoft Windows. Network-based locking at the file level provides a high level of concurrent access protection. NAS can also be optimized to

deliver file information to many clients with file-level protection. Two common applications that utilize the effectiveness of NAS include hosting home directories and providing a data store for static Web pages that are accessed by different Web servers. In certain situations, organizations can deploy NAS solutions for database applications in a limited manner. These situations are usually limited to applications for which the majority of data access is read-only, the databases are small, access volume is low, and predictable performance is not mandatory. In this type of situation, NAS solutions can reduce overall storage costs.

Careful evaluation of emerging trends in the networking and storage industry is imperative when selecting the appropriate technology. Along with the need for high performance, ease of management and sharing of data are also increasing. The choice of technology should balance these requirements; consider the complexity and maturity of the technology that is deployed. IP-SAN, detailed in the following chapter, is an emerging technology that has matured to meet enterprise demands.

EXERCISES

1. **List and explain the considerations for capacity design for both CPU and storage in a NAS environment.**
2. **SAN is configured for a backup to disk environment, and the storage configuration has additional capacity available. Can you have a NAS gateway configuration use this SAN? Discuss the implications of sharing the backup-to-disk SAN environment with NAS.**
3. **Explain how the performance of NAS can be affected if the TCP window size at the sender and the receiver are not synchronized.**
4. **Research the use of baby jumbo frames and how it affects NAS performance.**
5. **Research the file sharing features of the NFS protocol.**
6. **A NAS implementation configured jumbo frames on the NAS head, with 9,000 as its MTU. However, the implementers did not see any performance improvement and actually experienced performance degradation. What could be the cause? Research the end-to-end jumbo frame support requirements in a network.**
7. **Acme Corporation is trying to decide between an integrated or a gateway NAS solution. The existing SAN at Acme will provide capacity and scalability. The IT department is considering a NAS solution for the Training department at Acme for training videos. The videos would only be used by the training department for evaluation of instructors. Pick a NAS solution and explain the reasons for your choice.**