M.S. Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)
Department of Computer Science and Engineering

Course Name: Storage Area Networks

UNIT 4- Storage Networking Technologies

Network-Attached Storage(Chapter 7): Benefits of NAS, Components of NAS, NAS I/O Operation, NAS Implementations, NAS File Sharing Protocols, Factors Affecting NAS Performance, File-Level Virtualization.

Object Based and Unified Storage(Chapter 8): Object Based Storage Devices, Content Addressed Storage, CAS Use Cases, Unified Storage.

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

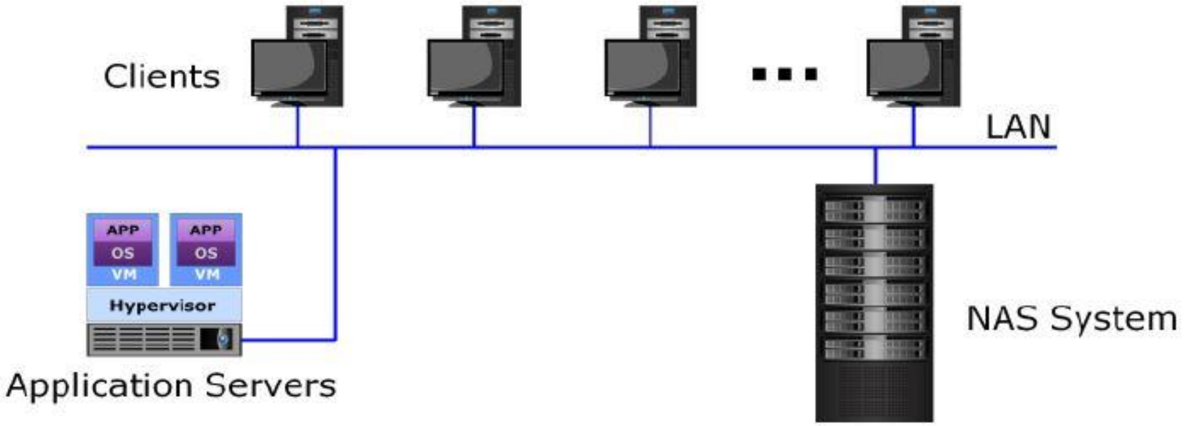
File-Level Virtualization

RAMAIAH Institute of Technology

Introduction

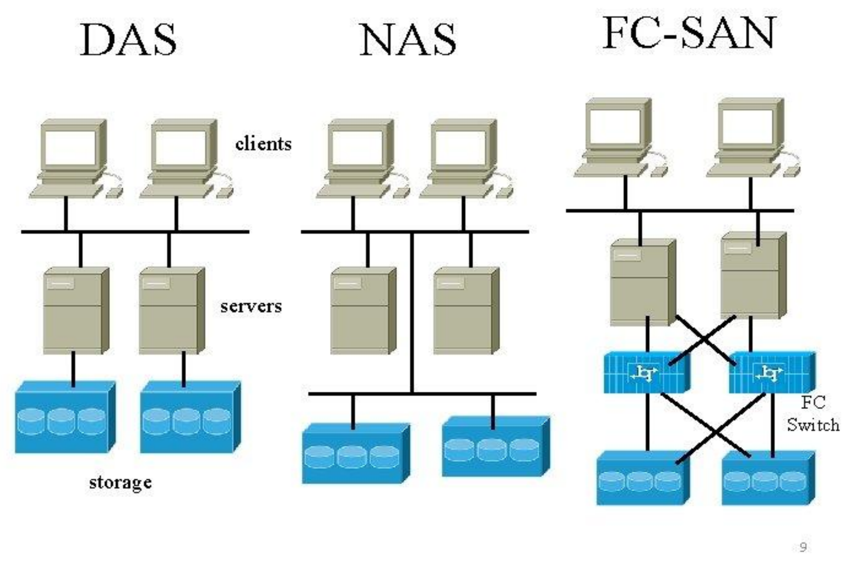
- **File sharing enables** users to share files with other users.
- Traditional methods of file sharing involve copying files to portable media such as floppy diskette, CD, DVD, or USB drives and delivering them to other users with whom it is being shared.
- However, this approach is not suitable in an enterprise environment in which a large number of users at different locations need access to common files.

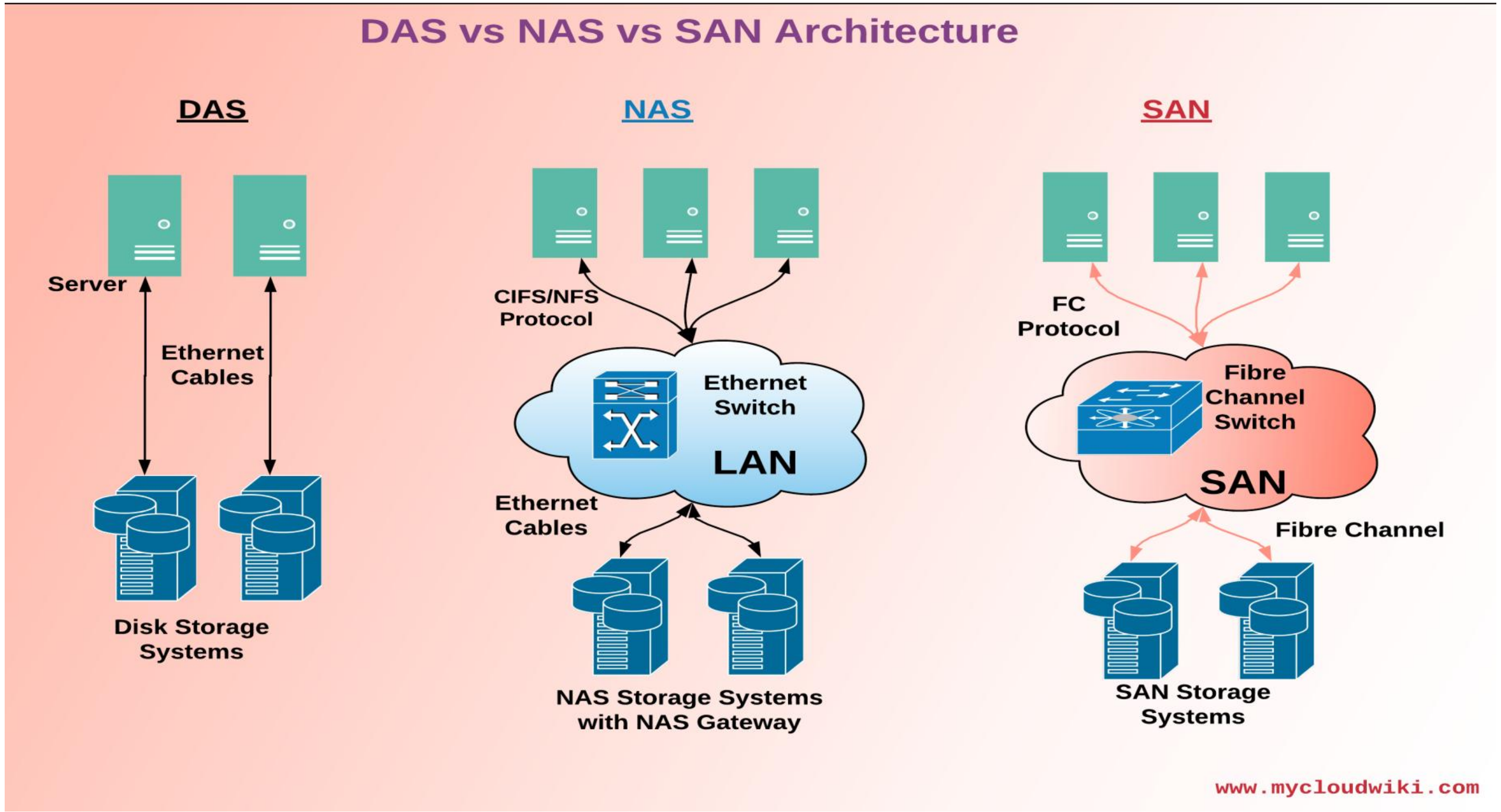
Introduction



- Network-based file sharing provides the flexibility to share files over long distances among a large number of users.
- File servers use client server technology to enable file sharing over a network.
- To address the tremendous growth of file data in enterprise environments, organizations have been deploying large numbers of file servers.
- These servers are either connected to direct-attached storage (DAS) or storage area network (SAN)-attached storage.

Introduction





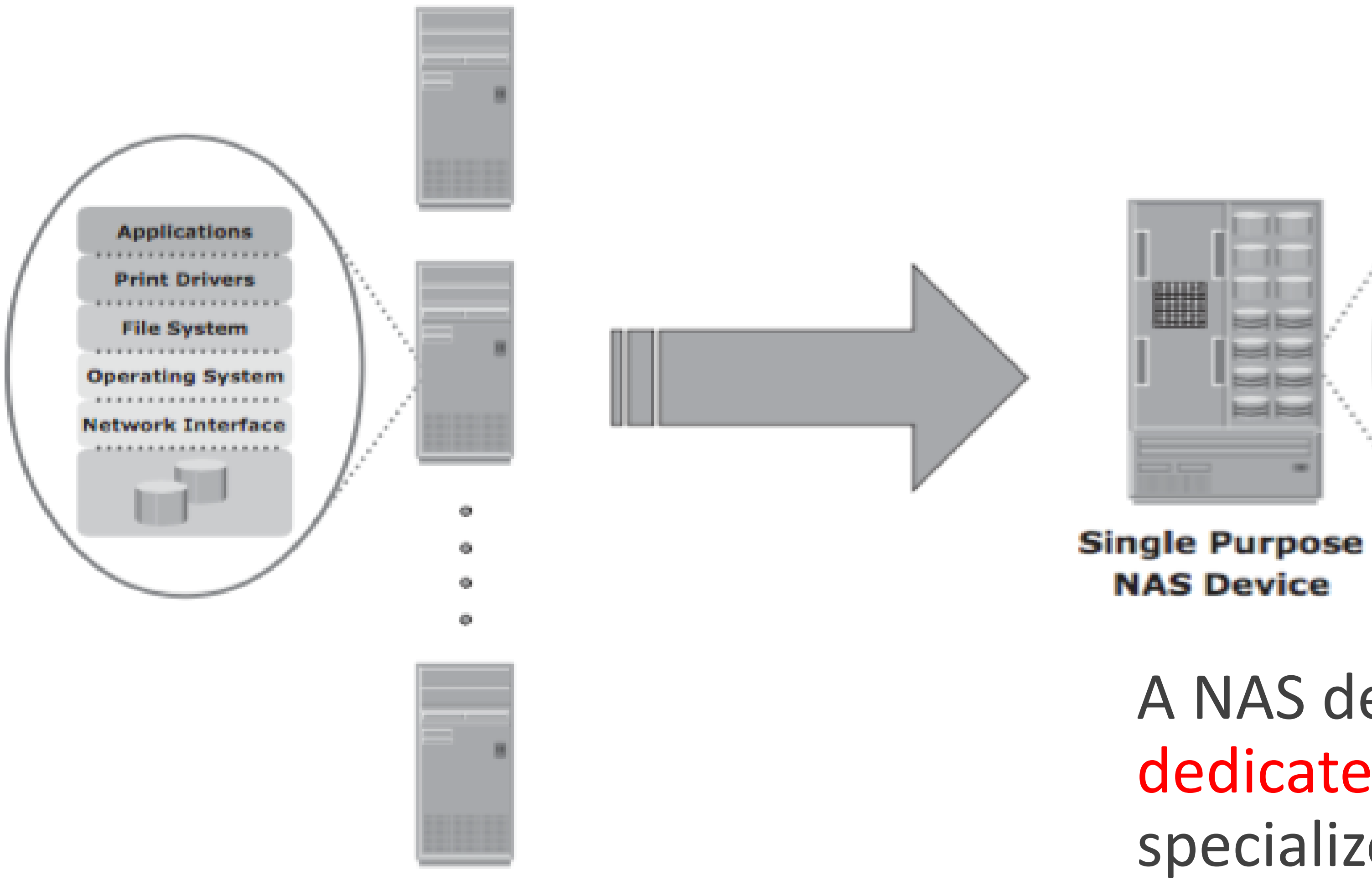


Figure 7-1: General purpose server versus NAS device

A NAS device is fundamentally a dedicated storage server -- a specialized computer designed and intended to support storage through network access.

RAMAIAH Institute of Technology

Introduction

- NAS is a dedicated, high-performance file sharing and storage device.
- A NAS device uses its own operating system and integrated hardware and software components to meet specific file-service needs.
- Its operating system is optimized for file I/O and performs file I/O better than a general-purpose server.
- NAS enables its clients to share files over an IP network.

Introduction

- NAS uses network and file-sharing protocols to provide access to the file data.

  These protocols include
  - TCP/IP for data transfer
  - Common Internet File System (CIFS)
  - Network File System (NFS) for network file service.

  NAS enables both UNIX and Microsoft Windows users to share the same data seamlessly.

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

File-Level Virtualization

Compare the benefits of Network-attached storage with General purpose server.

Benefits of NAS

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: NAS delivers better performance compared to a general-purpose file server because NAS uses an operating system specialized for file serving.

Benefits of NAS

Improved flexibility: Compatible with 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, and ensure greater data protection

Simplified management: Provides a centralized console that makes it possible to manage file systems efficiently

Benefits of NAS

Scalability: Scales well 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 components that provide maximum connectivity options. A NAS device supports clustering technology for failover.

Benefits of NAS

Security: Ensures security, user authentication, and file locking with industry-standard security schemas

Low cost: NAS uses commonly available and inexpensive Ethernet components.

Ease of deployment: Configuration at the client is minimal, because the clients have required NAS connection software built in.

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

File-Level Virtualization

Components of NAS

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

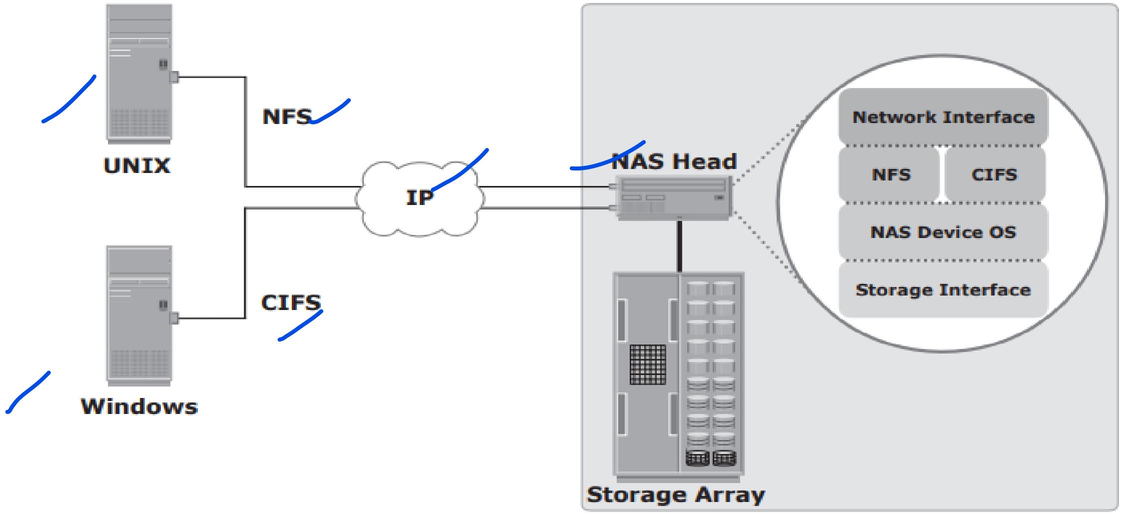
File-Level Virtualization

Explain the components of NAS, with neat diagram.

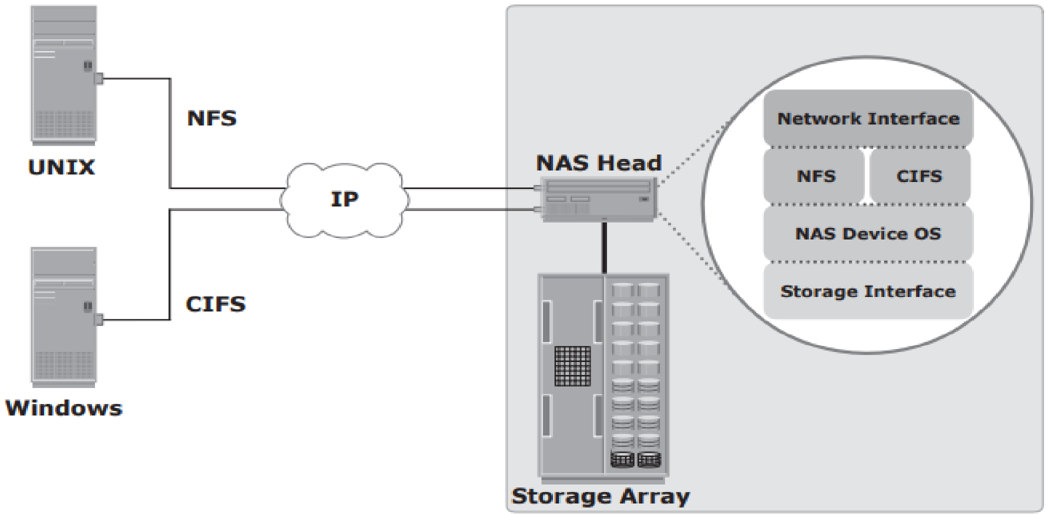
Components of NAS

A NAS device has two key components:

1. NAS head
2. Storage



Components of NAS



The NAS head includes the following components:

- CPU and memory
- One or more network interface cards (NICs), which provide connectivity to the client network.
- Examples of network protocols supported by NIC include Gigabit Ethernet, Fast Ethernet, ATM, and Fiber Distributed Data Interface (FDDI).
- An optimized operating system for managing the NAS functionality. It translates file-level requests into block-storage requests and further converts the data supplied at the block level to file data.
- Industry-standard storage protocols and ports to connect and manage physical disk resources

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

File-Level Virtualization

Explain the working of an I/O operation in NAS and list the benefits of using NAS.

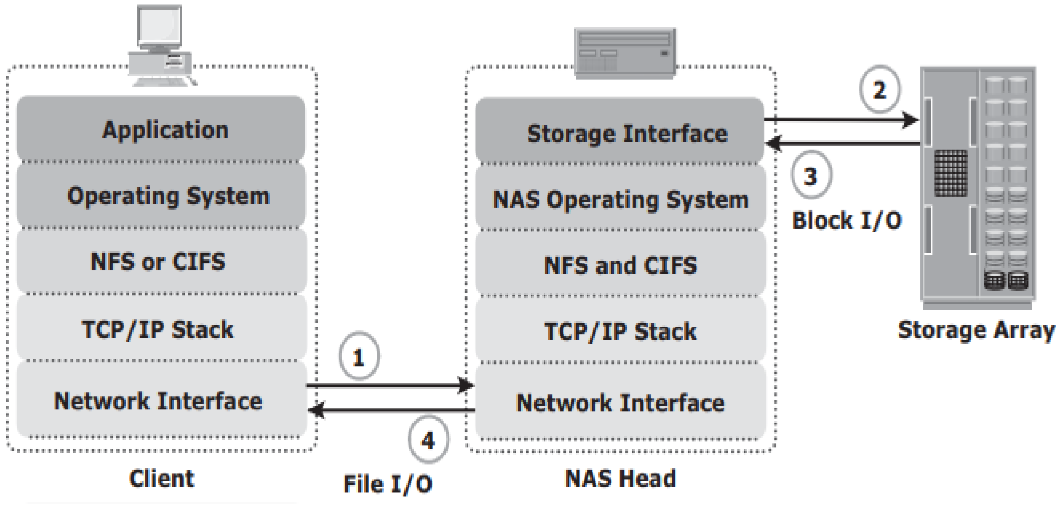
NAS I/O Operation

NAS provides file-level data access to its clients.

File I/O is a high-level request that specifies the file to be accessed.

For example, a client may request a file by specifying its name, location, or other attributes.

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.



RAMAIAH
Institute of Technology

NAS I/O Operation

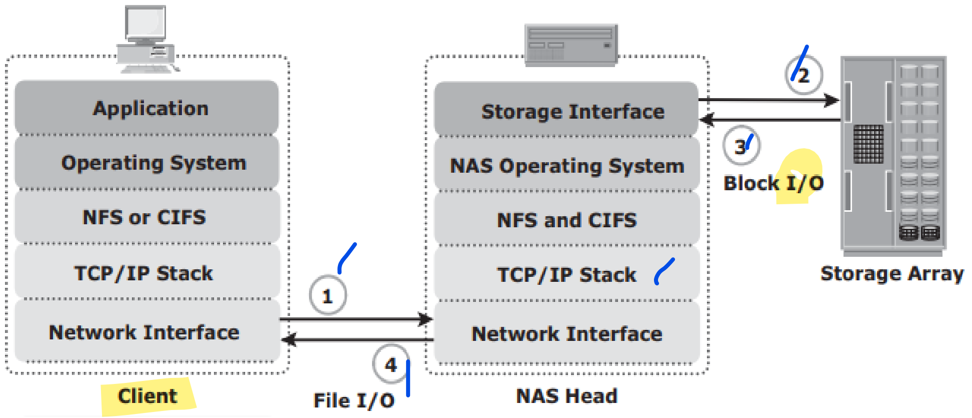


Figure 7-4: NAS I/O operation

The process of handling I/Os in a NAS environment is as follows:

1. The requestor (client) 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 on the physical storage.
3. When the NAS device receives data from the storage, it 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.

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

File-Level Virtualization

NAS Implementations

Three common NAS implementations are

1. Unified- NAS consolidates NAS-based and SAN-based data access within a unified storage platform and provides a unified management interface for managing both the environments.

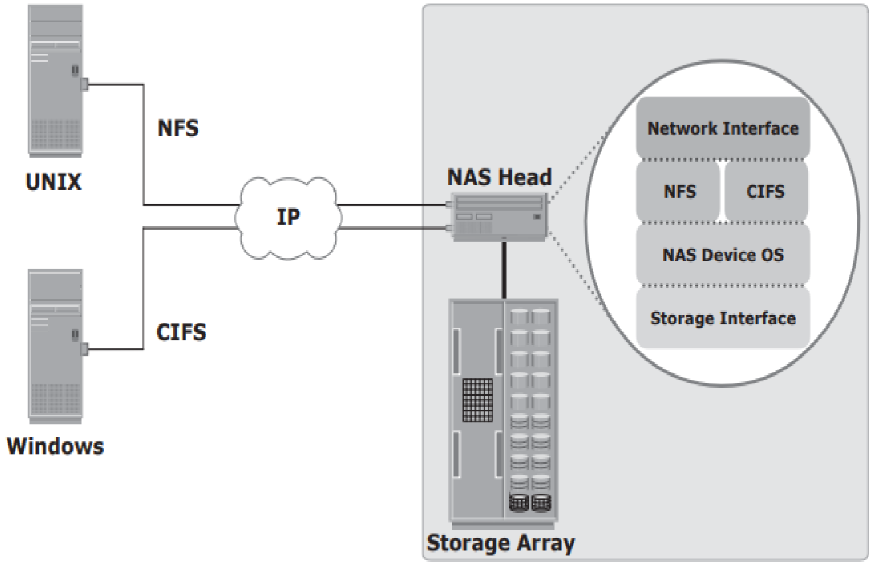
2. Gateway -NAS device uses external storage to store and retrieve data, and unlike unified storage, there are separate administrative tasks for the NAS device and storage.

3. scale-out-NAS implementation pools multiple nodes together in a cluster. A node may consist of either the NAS head or storage or both. The cluster performs the NAS operation as a single entity.

Unified NAS	Gateway NAS	Scale-out NAS
• Has **all** of its components and storage system in a **single** enclosure or frame	• NAS **head shares** its storage with SAN environment.	• Ideal for enterprise data centers • Consolidating both virtualized and non-virtualized file storage into **one storage pool** with a single point of management

NAS Implementations -Unified

- Unified NAS performs file serving and storing of file data, along with providing access to block-level data.
- It supports both CIFS and NFS protocols for file access and iSCSI and FC protocols for block level access.
- A unified NAS contains one or more NAS heads and storage in a single system.
- NAS heads are connected to the storage controllers (SCs), which provide access to the storage. These storage controllers also provide connectivity to iSCSI and FC hosts.
- The storage may consist of different drive types, such as SAS, ATA,FC, and flash drives, to meet different workload requirements.



NAS Implementations -Unified

Each NAS head in a unified NAS has front-end Ethernet ports, which connect to the IP network.

The front-end ports provide connectivity to the clients and service the file I/O requests.

Each NAS head has back-end ports, to provide connectivity to the storage controllers.

iSCSI and FC ports on a storage controller enable hosts to access the storage directly or through a storage network at the block level.

Figure 7-5 illustrates an example of unified NAS connectivity.

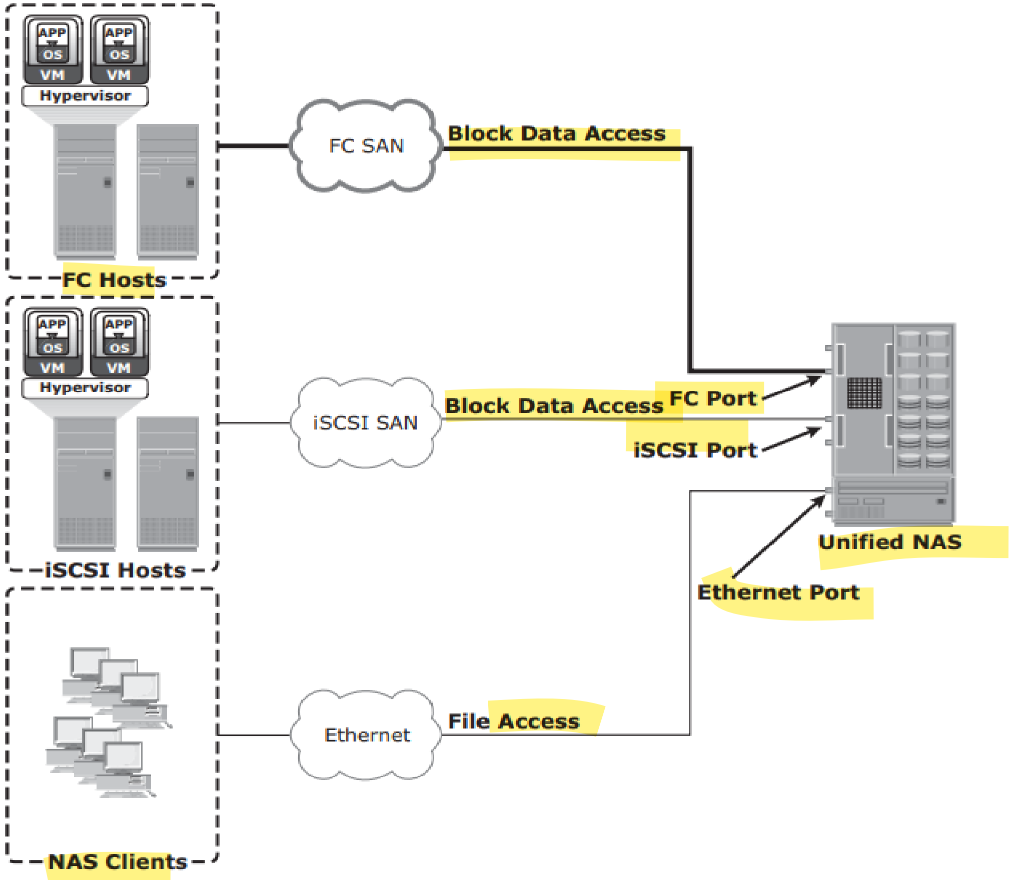


Figure 7-5: Unified NAS connectivity

NAS Implementations -Gateway

A gateway NAS device consists of one or more NAS heads and uses external and independently managed storage.

Management functions in this type of solution are more complex than those in a unified NAS environment because there are separate administrative tasks for the NAS head and the storage.

A gateway solution can use the FC infrastructure, such as switches and directors for accessing SAN-attached storage arrays or direct attached storage arrays.

The gateway NAS is more scalable compared to unified NAS because NAS heads and storage arrays can be independently scaled up when required.

RAMAIAH
Institute of Technology

NAS Implementations -Gateway

In a gateway solution, the front-end connectivity is similar to that in a unified storage solution.

Communication between the NAS gateway and the storage system in a gateway solution is achieved through a traditional FC SAN.

To deploy a gateway NAS solution, factors, such as multiple paths for data, redundant as a single entity. fabrics, and load distribution, must be considered.

Figure 7-6 illustrates an example of gateway NAS connectivity.

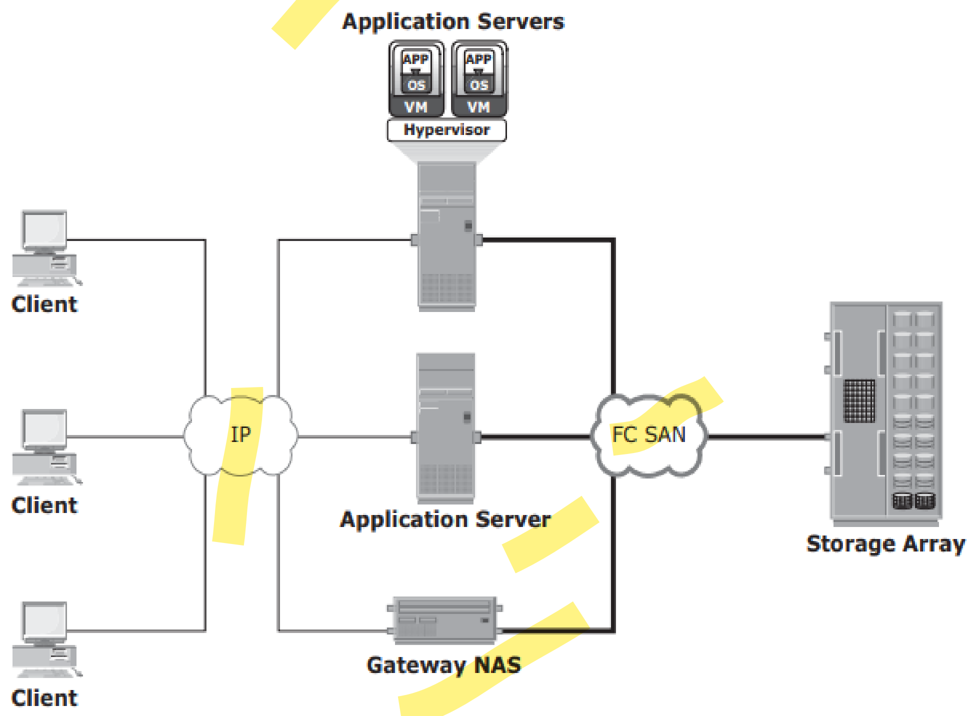


Figure 7-6: Gateway NAS connectivity

RAMAIAH
Institute of Technology

NAS Implementations -scale-out

Scale-out NAS clusters use separate internal and external networks for back-end and front-end connectivity, respectively.

An internal network provides connections for intracluster communication, and an external network connection enables clients to access and share fi le data.

Each node in the cluster connects to the internal network.

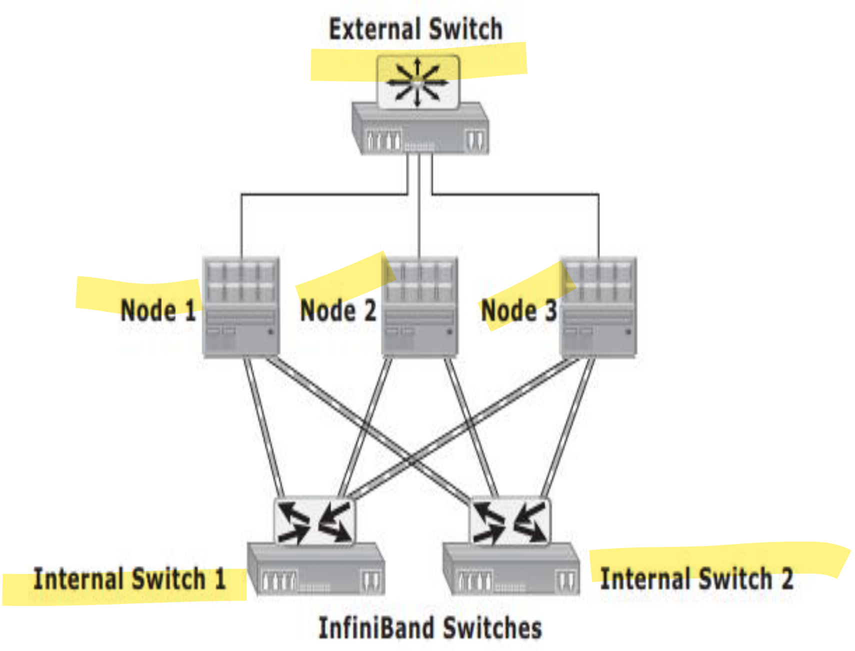


Figure 7-7: Scale-out NAS with dual internal and single external networks

NAS Implementations -scale-out

The internal network offers high throughput and low latency and uses high-speed networking technology, such as InfiniBand or Gigabit Ethernet.

To enable clients to access a node, the node must be connected to the external Ethernet network.

Redundant internal or external networks may be used for high availability.

Figure 7-7 illustrates an example of scale-out NAS connectivity.

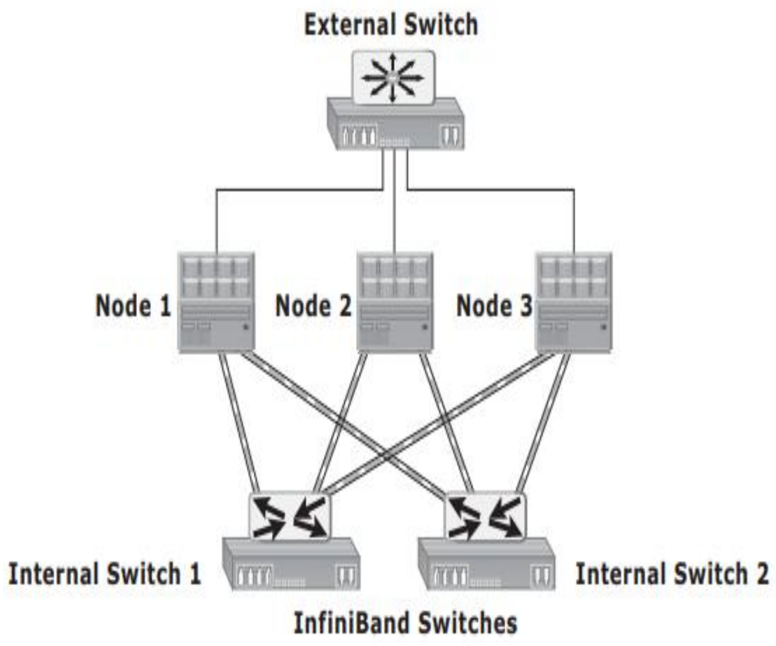


Figure 7-7: Scale-out NAS with dual internal and single external networks

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

File-Level Virtualization

NAS File Sharing Protocols

- Most NAS devices support multiple file-service protocols to handle file I/O requests to a remote file system.
- NFS and CIFS are the common protocols for file sharing.
- NAS devices 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.

NAS File Sharing Protocols - NFS

NFS is a client-server protocol for file sharing that is 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 inter-process communication between two computers.

NAS File Sharing Protocols - NFS

The NFS protocol provides a set of RPCs to access a remote fi le system for the following operations:

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

NAS File Sharing Protocols - NFS

NFS creates 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 handle reference to the file, a particular position to read or write, and the versions of NFS.

Ramaiah Institute of Technology

NAS File Sharing Protocols - 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, which uses UDP or TCP, and is based on the stateless protocol design. It includes some new features, such as a 64-bit fi le size, asynchronous writes, and additional file attributes to reduce refetching.
- **NFS version 4 (NFSv4):** Uses TCP and is based on a stateful protocol design. It offers enhanced security. The latest NFS version 4.1 is the enhancement of NFSv4 and includes some new features, such as session model, parallel NFS (pNFS), and data retention.

NAS File Sharing Protocols - 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 on a server.
- CIFS enables file sharing with other clients by using special locks.
- Filenames in CIFS are encoded using unicode characters.

NAS File Sharing Protocols - CIFS

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 fi le or a record.
- It supports fault tolerance and can automatically restore connections and reopen files that were open prior to an interruption.
  - The fault tolerance features of CIFS depend on whether an application is written to take advantage of these features.

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

File-Level Virtualization

Factors Affecting NAS Performance

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.

Authentication with a directory service such as Active Directory or NIS:

The authentication service must be available on the network with enough resources to accommodate the authentication load. Otherwise, a large number of authentication requests can increase latency.

RAMAIAH Institute of Technology

Factors Affecting NAS Performance

Retransmission:

- Link errors and buffer overflows can result in retransmission.
- This causes packets that have not reached the specified destination to be re-sent.
- Care must be taken to match both speed and duplex settings on the network devices and the NAS heads.
- Improper configuration might result in errors and retransmission, adding to latency.

RAMAIAH Institute of Technology

Factors Affecting NAS Performance

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 utilization statistics to determine the optimum utilization of switches and routers in a network.

Additional devices should be added if the current devices are overutilized.

RAMAIAH
Institute of Technology

Factors Affecting NAS Performance

File system lookup and metadata requests: NAS clients access files on NAS devices. The processing required to reach 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.

Over utilized NAS devices: Clients accessing multiple fi les can cause high utilization levels on a NAS device, which can be determined by viewing utilization statistics. High memory, CPU, or disk subsystem utilization levels can be caused by a poor fi le system structure or insufficient resources in a storage subsystem.

RAMAIAH
Institute of Technology

Factors Affecting NAS Performance

Over utilized clients:

- The client accessing CIFS or NFS data might also be over utilized.
- An overutilized client requires a longer time to process the requests and responses.
- Specific performance-monitoring tools are available for various operating systems to help determine the utilization of client resources.

Network-Attached Storage(Chapter 7)

Benefits of NAS

Components of NAS

NAS I/O Operation

NAS Implementations

NAS File Sharing Protocols

Factors Affecting NAS Performance

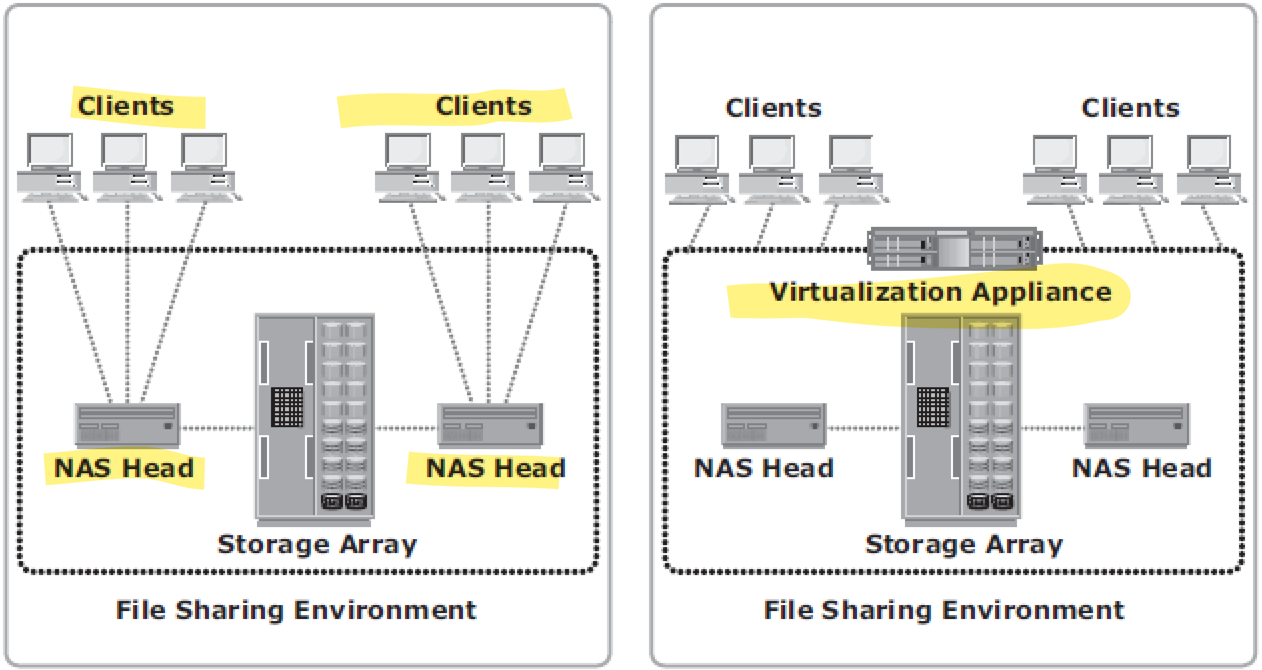
File-Level Virtualization

Network-Attached Storage(Chapter 7)

File-level virtualization eliminates the dependencies between the data accessed at the file level and the location where the files are physically stored.

Implementation of file-level virtualization is common in NAS or file-server environments.

It provides non-disruptive file mobility to optimize storage utilization.



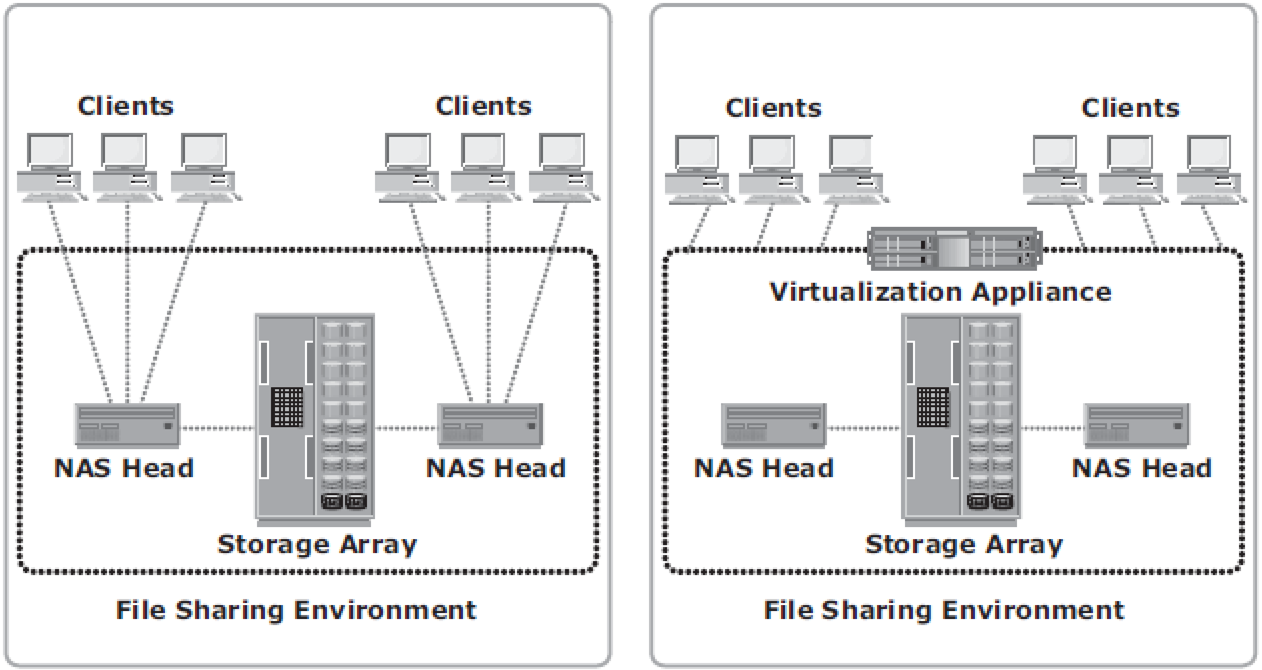
(a) Before File-Level Virtualization (b) After File-Level Virtualization

Network-Attached Storage(Chapter 7)

Before virtualization, each host knows exactly where its file resources are located.

This environment leads to underutilized storage resources and capacity problems because files are bound to a specific NAS device or file server.

It may be required to move the files from one server to another because of performance reasons or when the file server fills up.



(a) Before File-Level Virtualization (b) After File-Level Virtualization

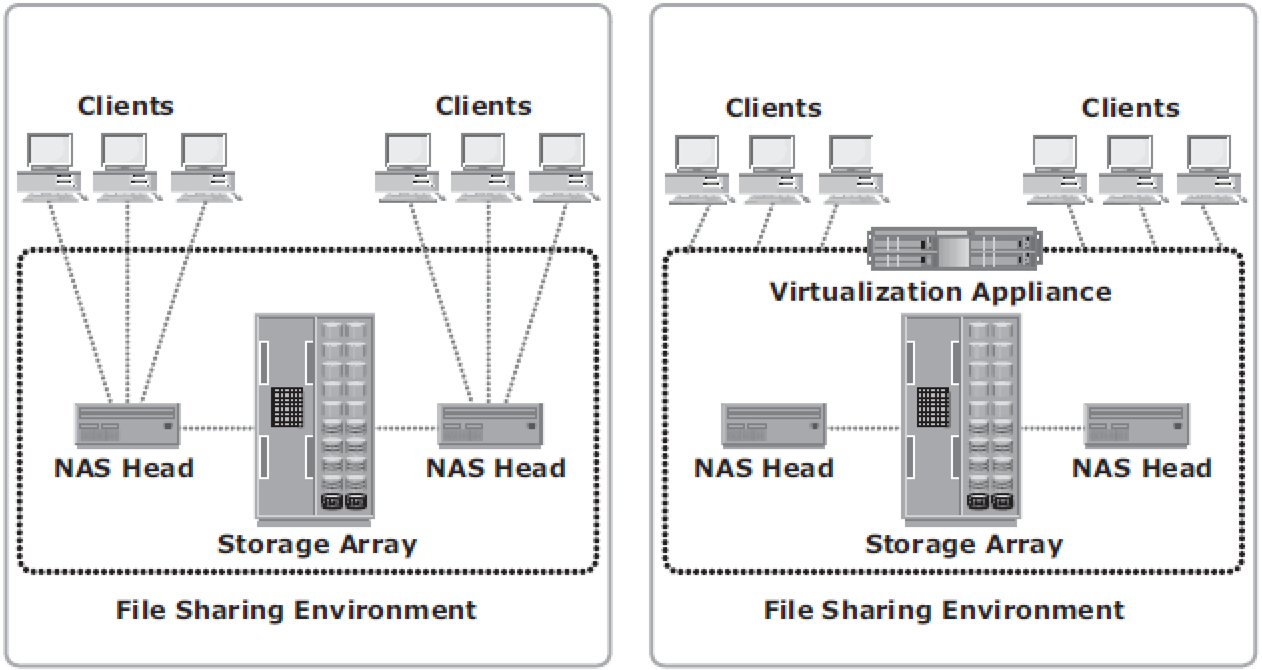
Network-Attached Storage(Chapter 7)

File-level virtualization simplifies file mobility.

It provides user or application independence from the location where the files are stored.

File-level virtualization creates a logical pool of storage, enabling users to use a logical path, rather than a physical path, to access files.

File-level virtualization facilitates the movement of fi les across the online file servers or NAS devices.



(a) Before File-Level Virtualization (b) After File-Level Virtualization

UNIT 4- Storage Networking Technologies

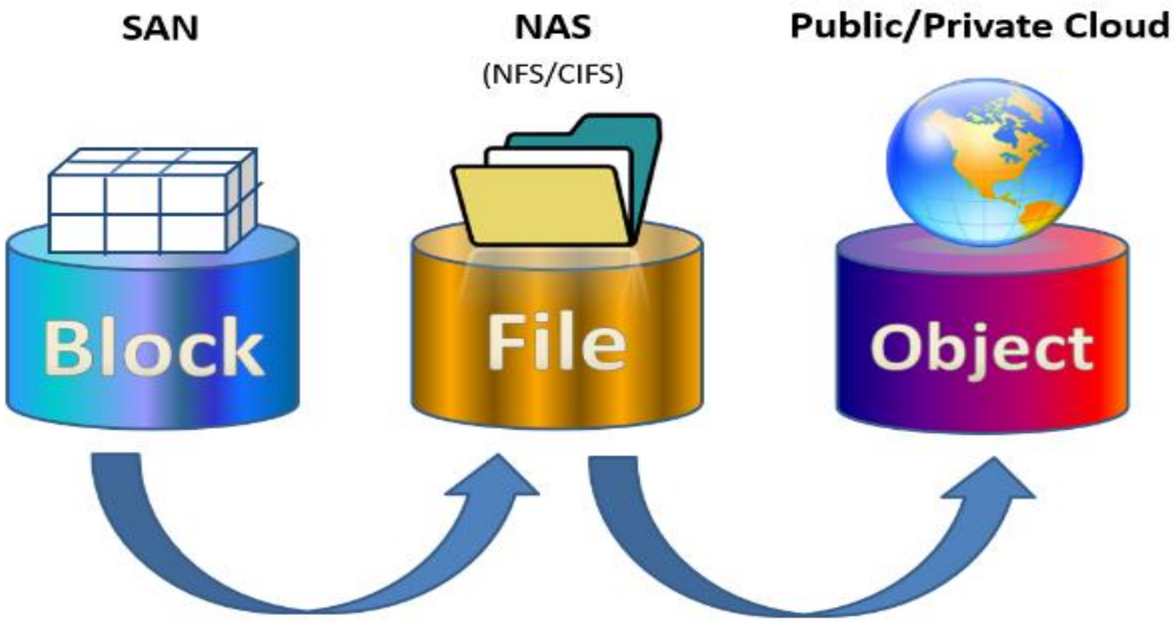
Object Based and Unified Storage(Chapter 8):

Object Based Storage Devices

Content Addressed Storage

CAS Use Cases

Unified Storage



Introduction

- 90 percent of data generated is unstructured.
- NAS is a dominant solution for storing unstructured data, has become inefficient.
- Data growth adds high overhead to the network-attached storage (NAS) in terms of managing a large number of permissions and nested directories.

Smarter approach is needed to manage unstructured data based on its content.

Object-based storage is a way to store file data in the form of objects based on its content and other attributes.

Describe the process of storing and retrieving objects in an Object-Based Storage Device (OBSD).

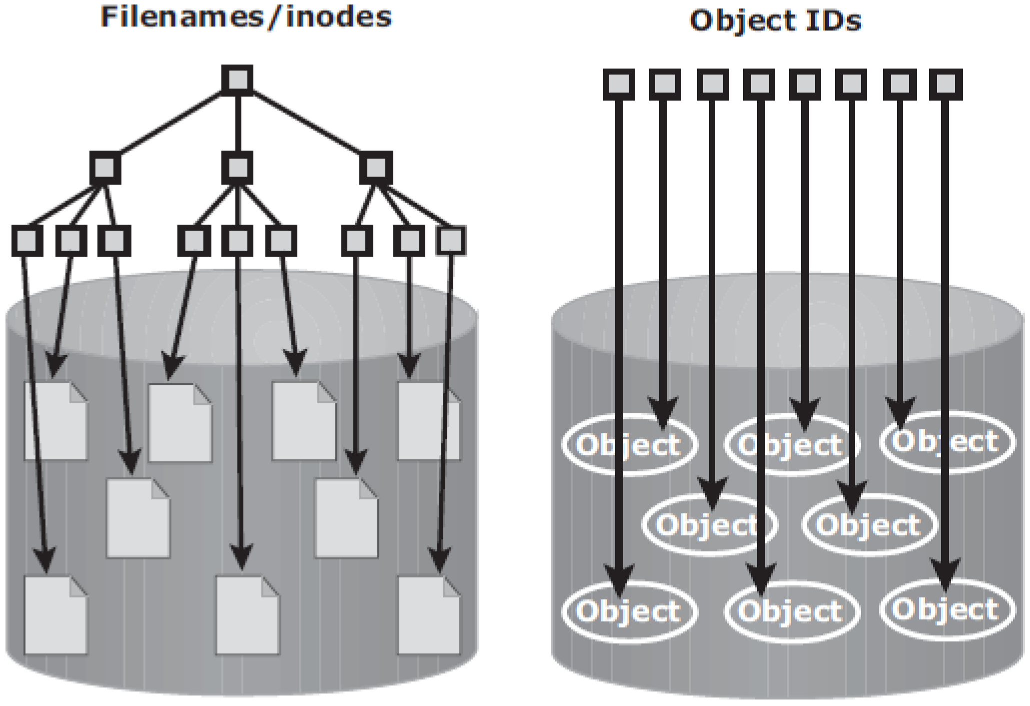
Object Based Storage Devices

An OSD is a device that organizes and stores unstructured data as objects.

OSD stores data in the form of objects.

OSD uses flat address space to store data.

Therefore, there is no hierarchy of directories and files; as a result, a large number of objects can be stored in an OSD system (see Figure 8-1).



(a) Hierarchical File System (b) Flat Address Space

Object Based Storage Devices

An object might contain user data, related metadata (size, date, ownership), and other attributes of data (retention, access pattern, and so on); see Figure 8-2.

Each object stored in the system is identified by a unique ID called the object ID.

The object ID is generated using specialized algorithms such as hash function on the data and guarantees that every object is uniquely identified.

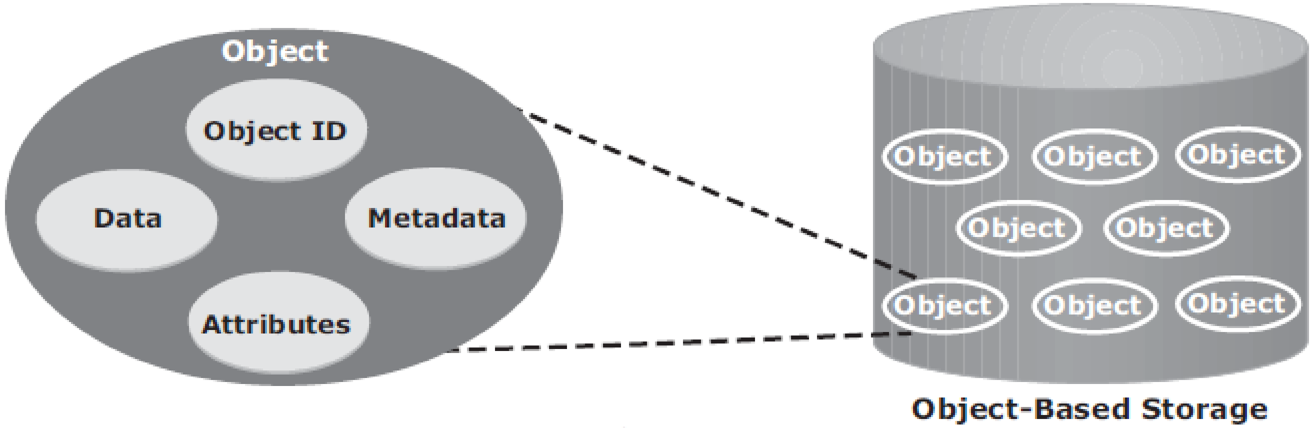
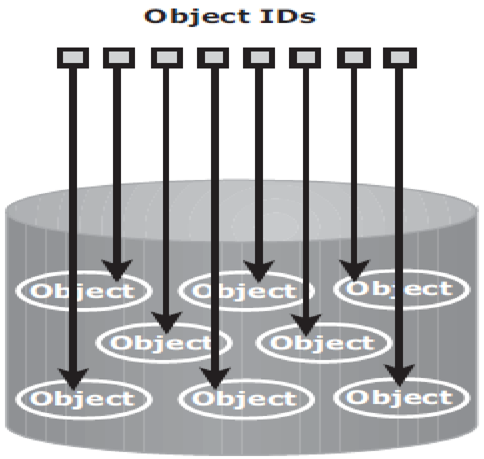


Figure 8-2: Object structure



Object Based Storage Devices

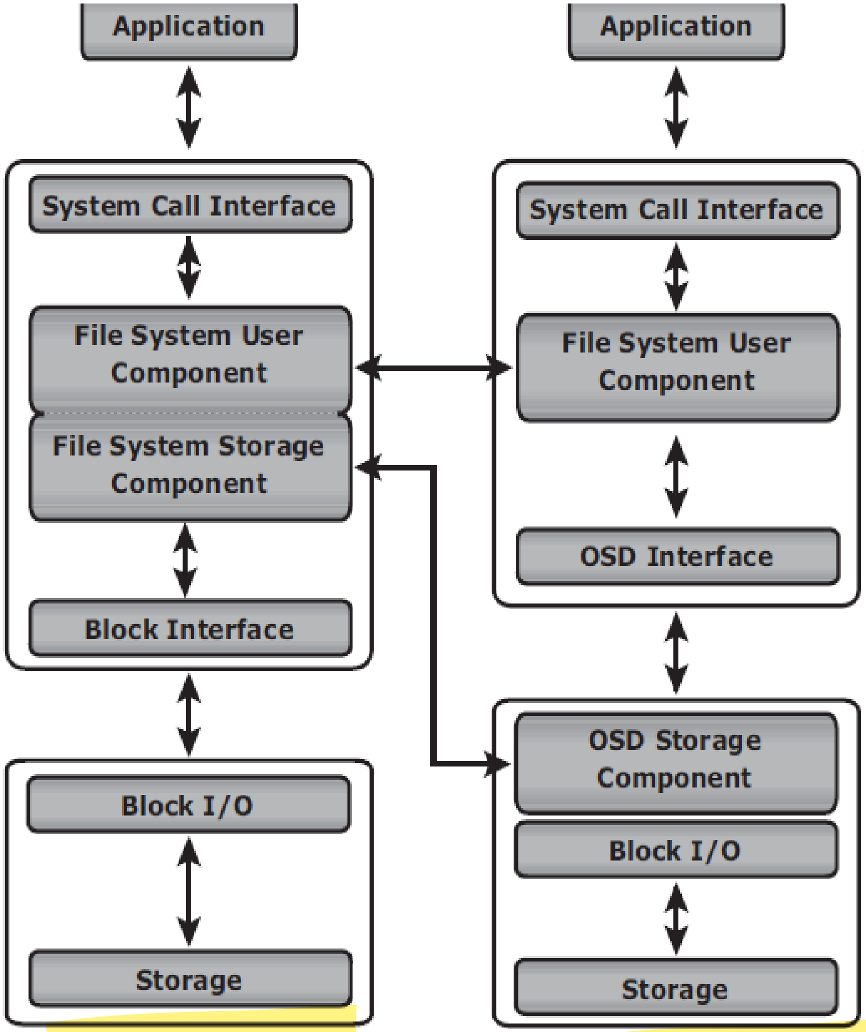
Object-Based Storage Architecture

Figure 8-3 (a) illustrates the **Block-level access**.

The I/O generated by an application passes through the file system, the channel, or network and reaches the disk drive.

When the file system receives the I/O from an application, the file system maps the incoming I/O to the disk blocks. The block interface is used for sending the I/O over the channel or network to the storage device.

The I/O is then written to the block allocated on the disk drive.



(a) Block-Level Access (b) Object-Level Access

Object Based Storage Devices

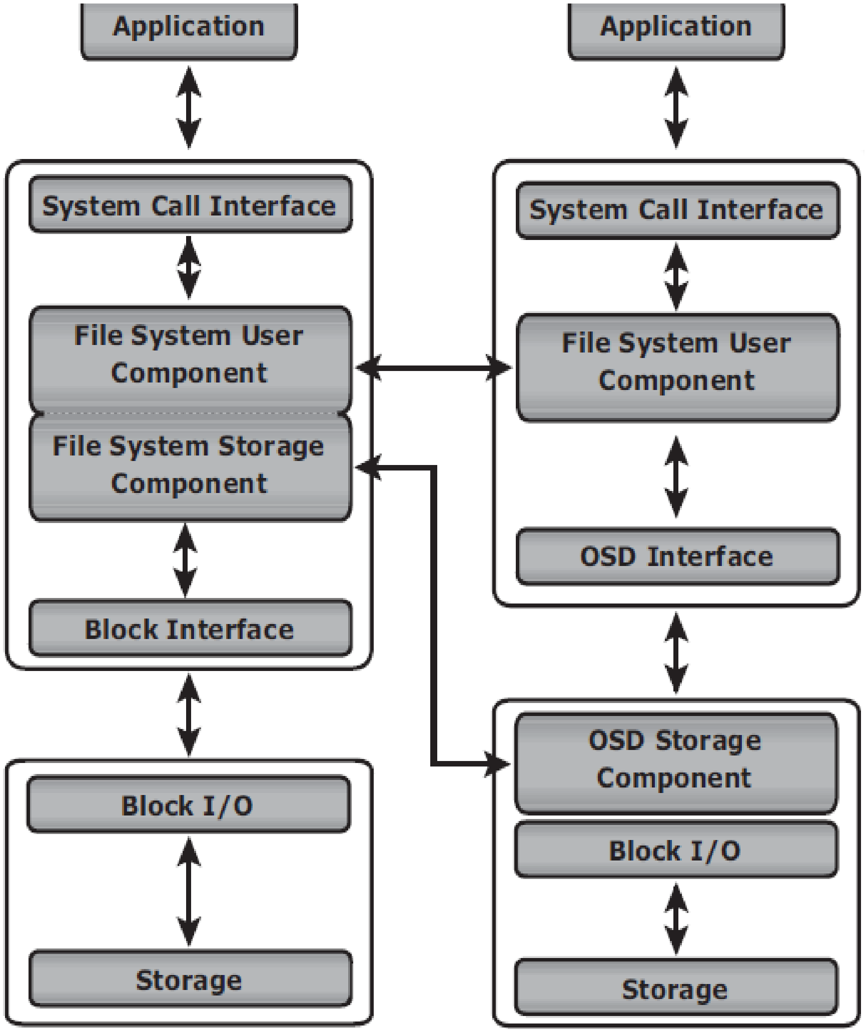
Object-Based Storage Architecture

The file system has two components:

1. user component
2. storage component.

The user component of the file system performs functions such as hierarchy management, naming, and user access control.

The storage component maps the files to the physical location on the disk drive.



(a) Block-Level Access (b) Object-Level Access

Object Based Storage Devices

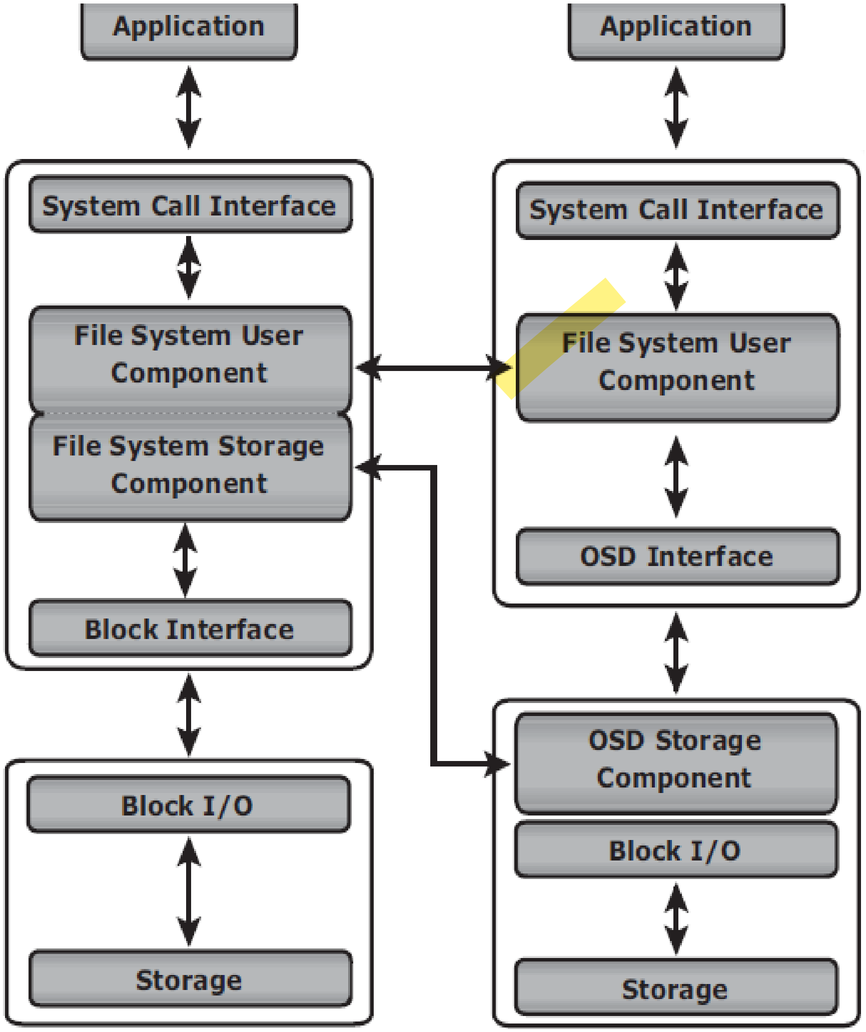
Object-Based Storage Architecture

Figure 8-3 (b) illustrates the Object-level access

When an application accesses data stored in OSD, the request is sent to the file system user component.

The file system user component communicates to the OSD interface, which in turn sends the request to the storage device.

The storage device has the OSD storage component responsible for managing the access to the object on a storage device.



(a) Block-Level Access (b) Object-Level Access

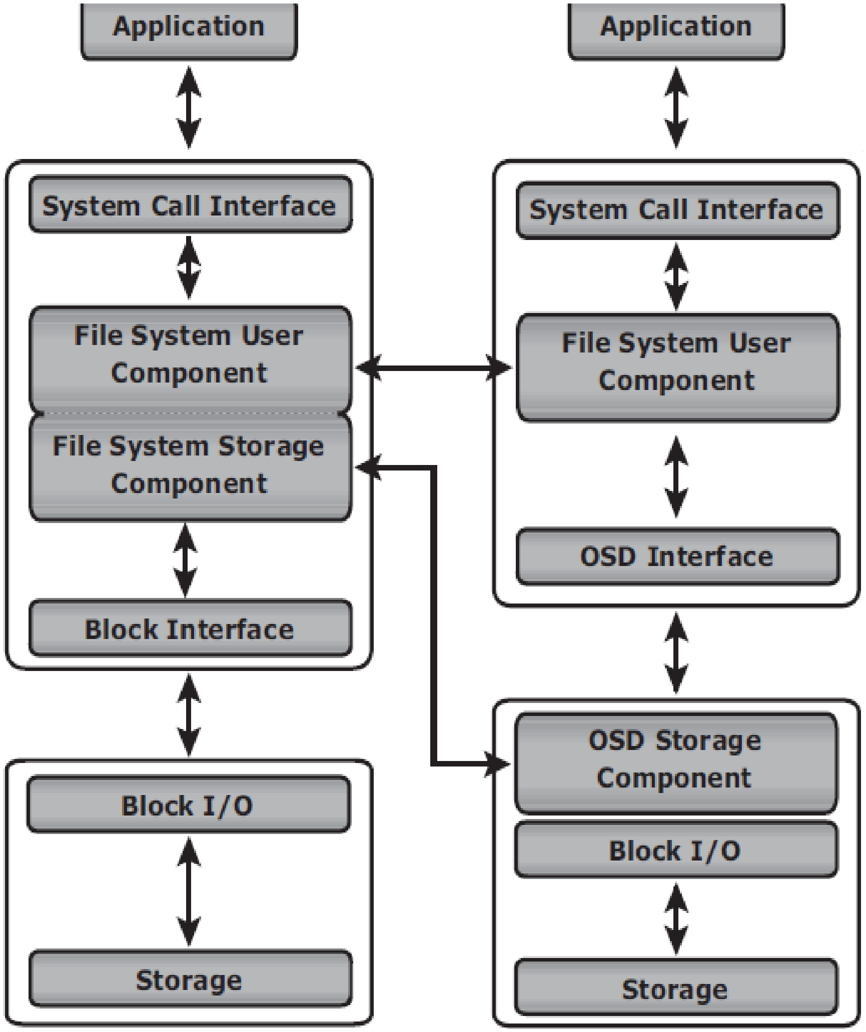
Object Based Storage Devices

Object-Based Storage Architecture

After the object is stored, the OSD sends an acknowledgment to the application server.

The OSD storage component manages all the required low-level storage and space management functions.

It also manages security and access control functions for the objects.



(a) Block-Level Access (b) Object-Level Access

Describe the components of an Object-Based Storage Device (OSD) with the aid of a neat diagram.

Object Based Storage Devices

Components of OSD

The OSD system is typically composed of three key components:

1. Nodes
2. Private network
3. Storage

Figure 8-4 illustrates the components of OSD.

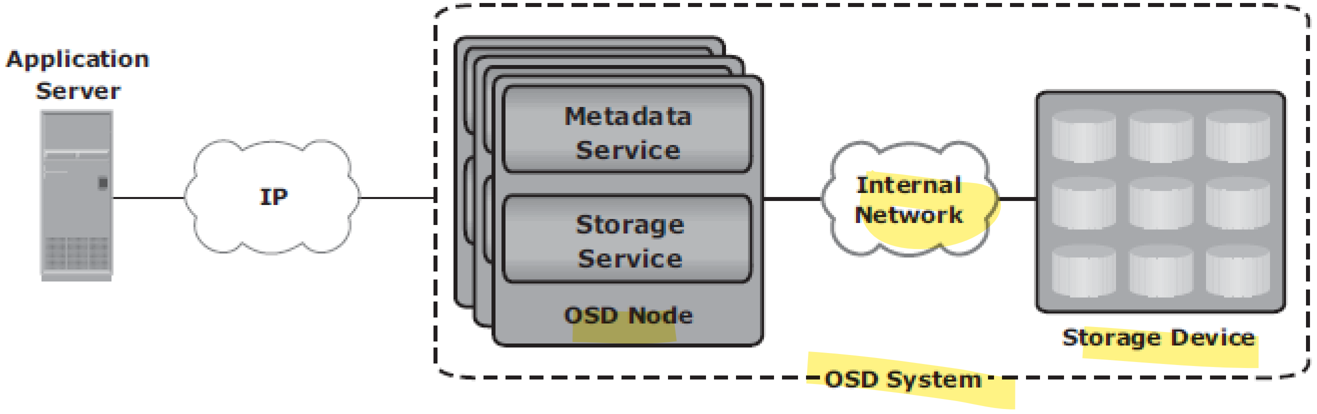


Figure 8-4: OSD components

The OSD system is composed of one or more nodes.

Object Based Storage Devices

Components of OSD

A Node is a server that runs the OSD operating environment and provides services to store, retrieve, and manage data in the system.

The OSD node has two key services:

1. metadata service
2. storage service.

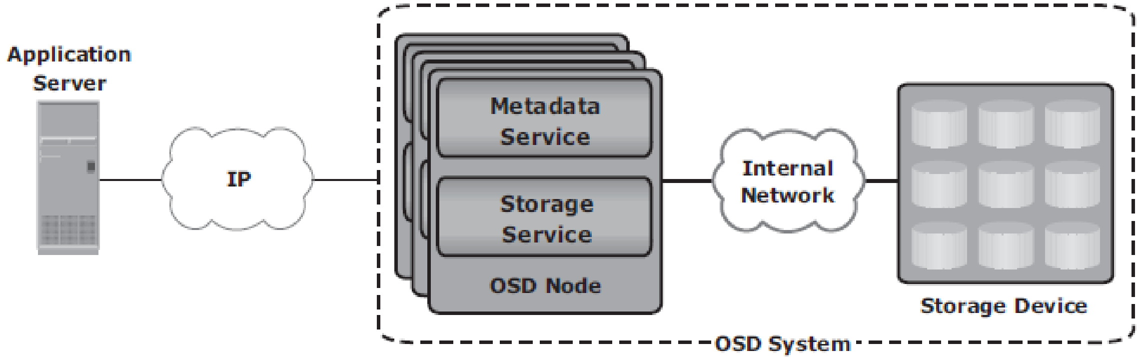


Figure 8-4: OSD components

Object Based Storage Devices

Components of OSD

- The metadata service is responsible for generating the object ID from the contents of a file. It also maintains the mapping of the object IDs and the file system namespace.
- The storage service manages a set of disks on which the user data is stored.
- The OSD nodes connect to the storage via an internal network.
- The internal network provides node-to-node connectivity and node-to-storage connectivity.

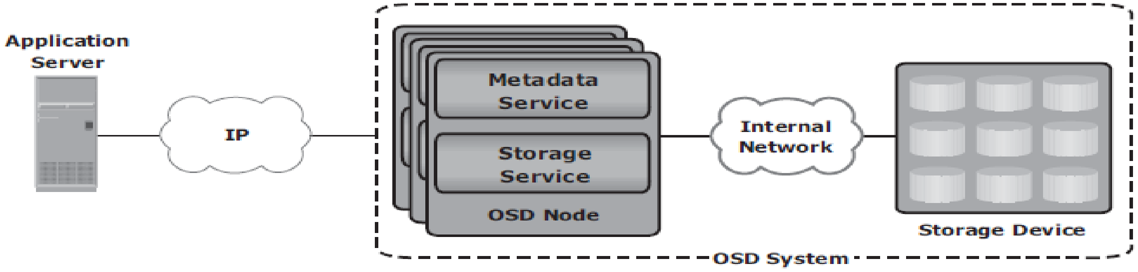


Figure 8-4: OSD components

Object Based Storage Devices

Object Storage in OSD

The data storage process in an OSD system is as follows:

1. The application server presents the file to be stored to the OSD node.

2. The OSD node divides the file into two parts: user data and metadata.

3. The OSD node generates the object ID using a specialized algorithm. The algorithm is executed against the contents of the user data to derive an ID unique to this data.

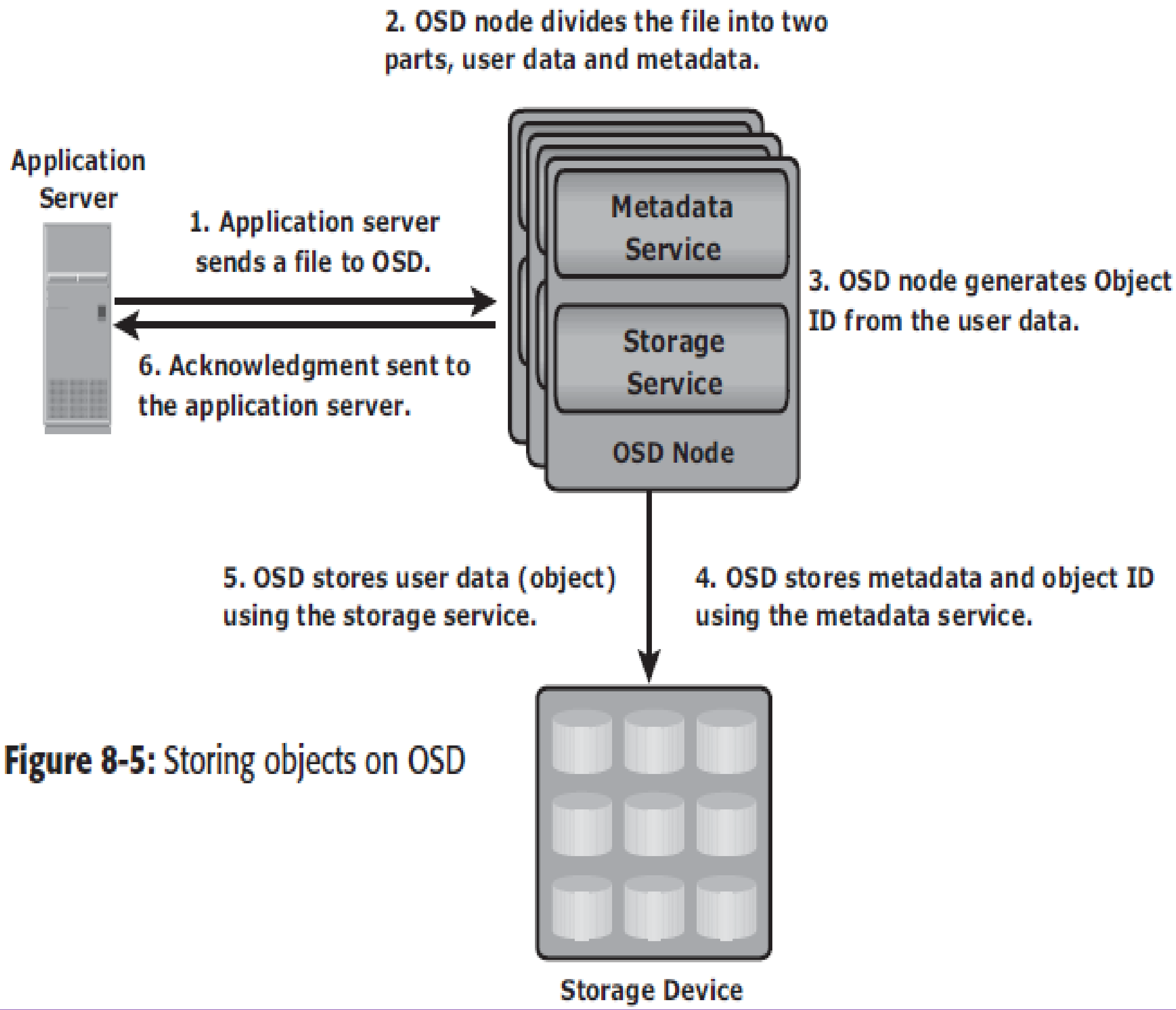


Figure 8-5: Storing objects on OSD

Object Based Storage Devices

Object Storage in OSD

4. For future access, the OSD node stores the metadata and object ID using the metadata service.

5. The OSD node stores the user data (objects) in the storage device using the storage service.

6. An acknowledgment is sent to the application server stating that the object is stored.

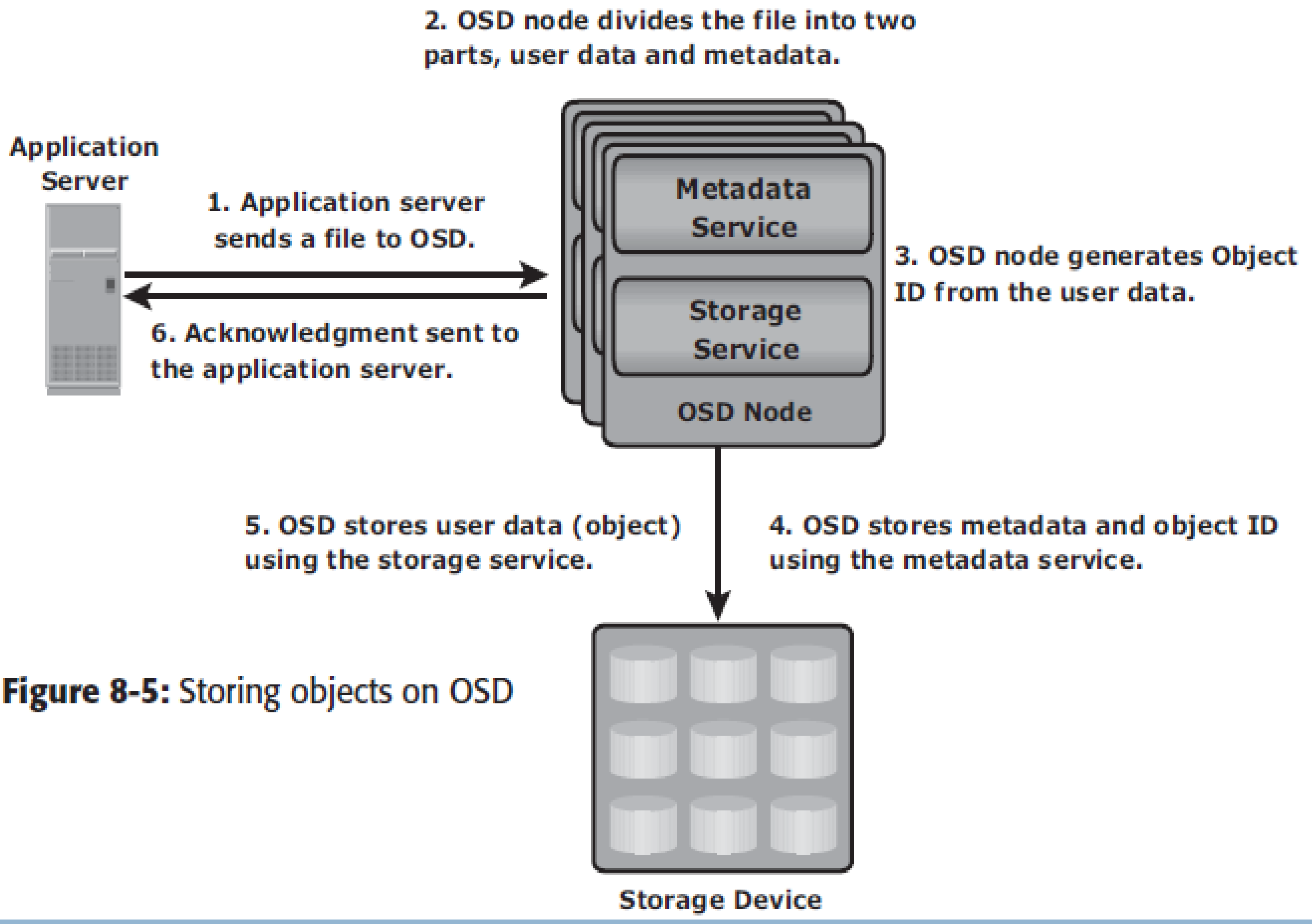


Figure 8-5: Storing objects on OSD

Object Based Storage Devices

Object Retrieval in OSD

1. The application server sends a read request to the OSD system.

2. The metadata service retrieves the object ID for the requested file.

3. The metadata service sends the object ID to the application server.

4. The application server sends the object ID to the OSD storage service for object retrieval.

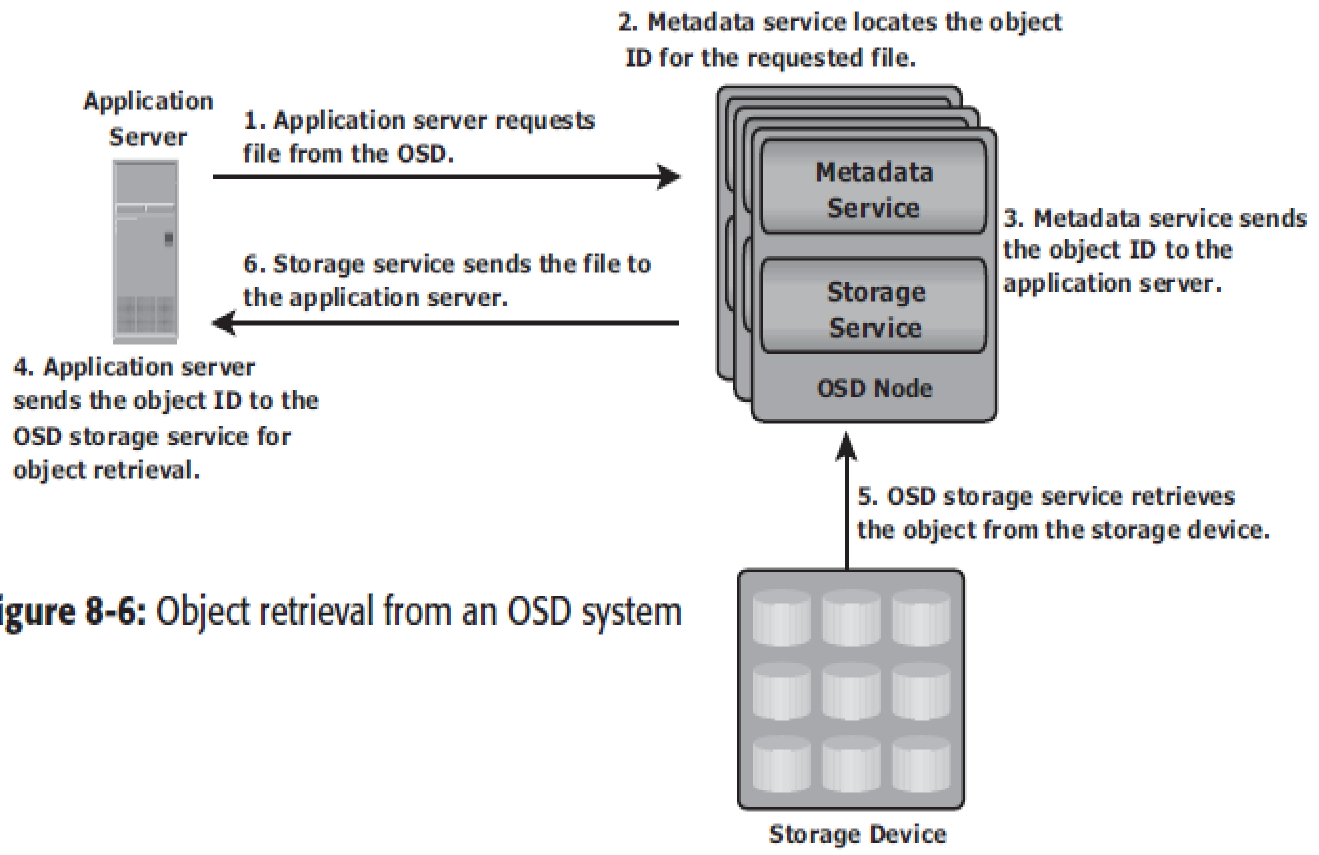


Figure 8-6: Object retrieval from an OSD system

RAMAIAH
Institute of Technology

Object Based Storage Devices

Object Retrieval in OSD

5. The OSD storage service retrieves the object from the storage device.

6. The OSD storage service sends the file to the application server.

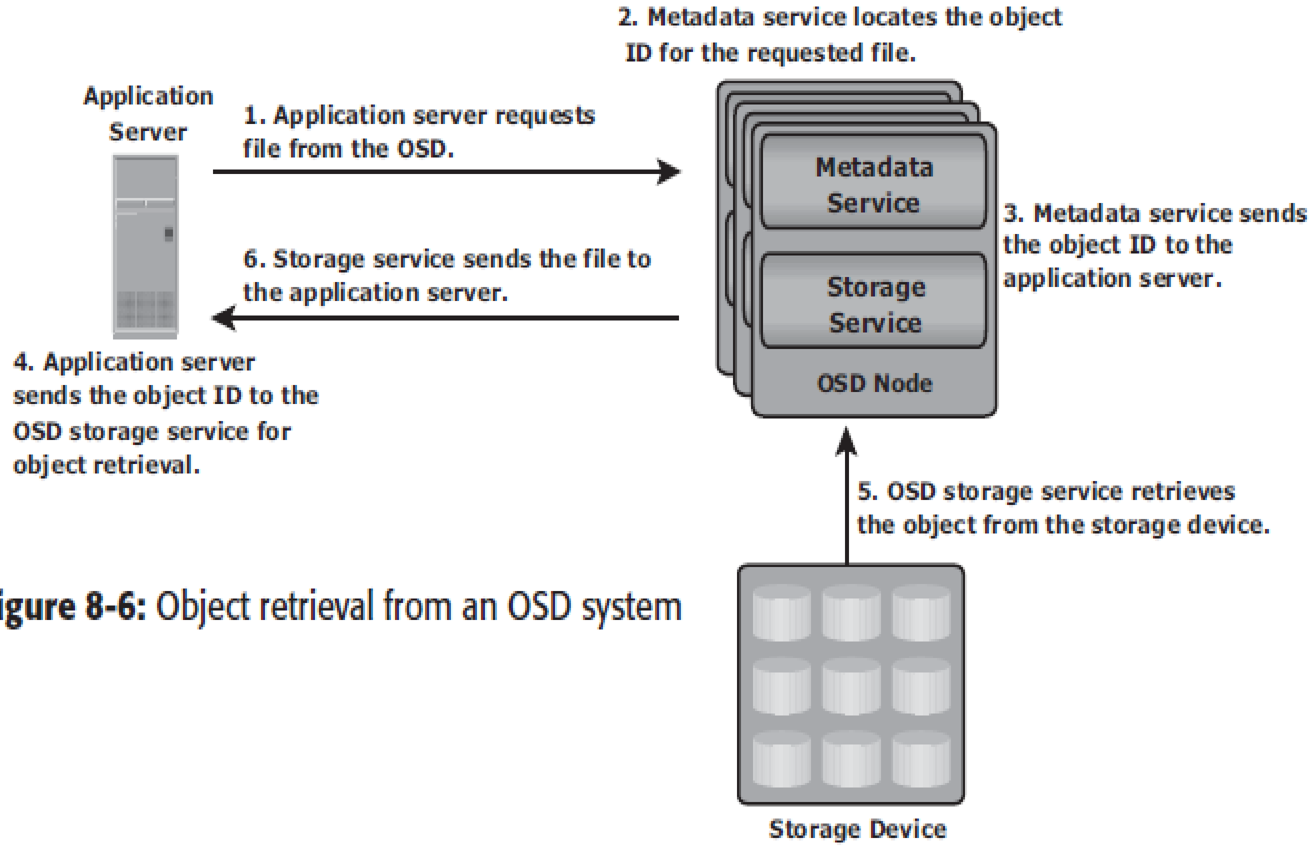


Figure 8-6: Object retrieval from an OSD system

RAMAIAH
Institute of Technology

Object Based Storage Devices

Benefits of Object-Based Storage

1. Security and reliability
2. Platform independence
3. Scalability
4. Manageability

UNIT 4- Storage Networking Technologies

Object Based and Unified Storage(Chapter 8):

Object Based Storage Devices

Content Addressed Storage

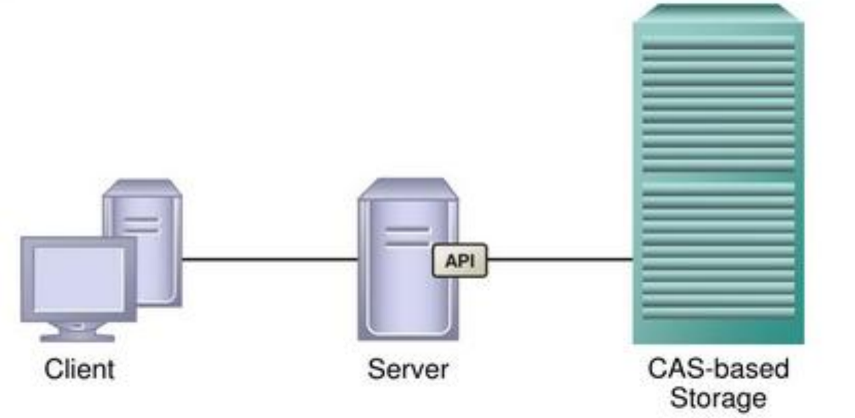
CAS Use Cases

Unified Storage

Explain why Content Addressed Storage (CAS) features are essential for storing fixed content effectively.

Content Addressed Storage(CAS)

- CAS is an object-based storage device designed for secure online storage and retrieval of fixed content.
- CAS stores user data and its attributes as an object.
- The stored object is assigned a globally unique address, known as a content address (CA).
- This address is derived from the object's binary representation.
- CAS provides an optimized and centrally managed storage solution. Data access in CAS differs from other OSD devices.



Content Addressed Storage(CAS)

The key features of CAS are as follows:

Content integrity: It provides assurance that the stored content has not been altered.

Location independence: CAS uses a unique content address, rather than directory path names or URLs, to retrieve data.

Content Addressed Storage(CAS)

The key features of CAS are as follows:

Single-instance storage (SIS): CAS uses a unique content address to guarantee the storage of only a single instance of an object.

Retention enforcement: Protecting and retaining objects is a core requirement of an archive storage system.

Data protection: CAS ensures that the content stored on the CAS system is available even if a disk or a node fails.

Fast record retrieval: CAS stores all objects on disks, which provides faster access to the objects compared to tapes and optical discs.

Content Addressed Storage(CAS)

The key features of CAS are as follows:

Load balancing: CAS distributes objects across multiple nodes to provide maximum throughput and availability.

Scalability: CAS allows the addition of more nodes to the cluster without any interruption to data access and with minimum administrative overhead.

Content Addressed Storage(CAS)

The key features of CAS are as follows:

Event notification: CAS continuously monitors the state of the system and raises an alert for any event that requires the administrator's attention.

Self diagnosis and repair: CAS automatically detects and repairs corrupted objects and alerts the administrator about the potential problem.

Audit trails: CAS keeps track of management activities and any access or disposition of data. Audit trails are mandated by compliance requirements.

UNIT 4- Storage Networking Technologies

Object Based and Unified Storage(Chapter 8):

Object Based Storage Devices,

Content Addressed Storage,

CAS Use Cases,

Unified Storage.

The IT department of a department store currently uses tape for data archiving. What are four to five key points you would present to persuade them to adopt a CAS solution instead? How would implementing these suggestions impact the IT department?

CAS Use Cases

Healthcare Solution: Storing Patient Studies

Figure 8-7 illustrates CAS in this scenario.

Patients' records are retained on the primary storage for 60 days after which they are moved to the CAS system.

CAS facilitates long-term storage and at the same time, provides immediate access to data, when needed.

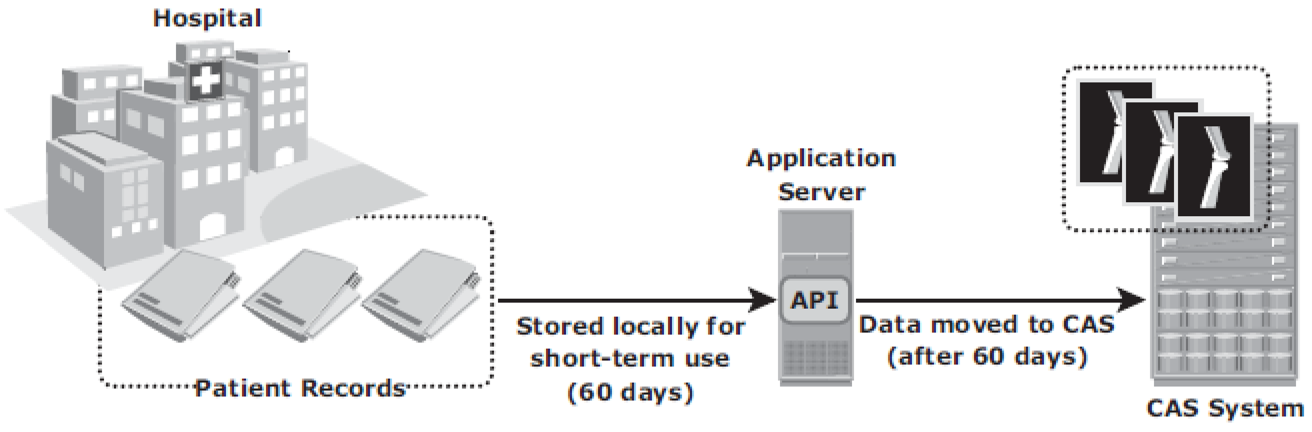


Figure 8-7: Storing patient studies on a CAS system

CAS Use Cases

Finance Solution: Storing Financial Records

Figure 8-8 illustrates the use of CAS in this scenario.

The check images are moved from the primary storage to the CAS system after 60 days, and can be held there for long term based on retention policy.

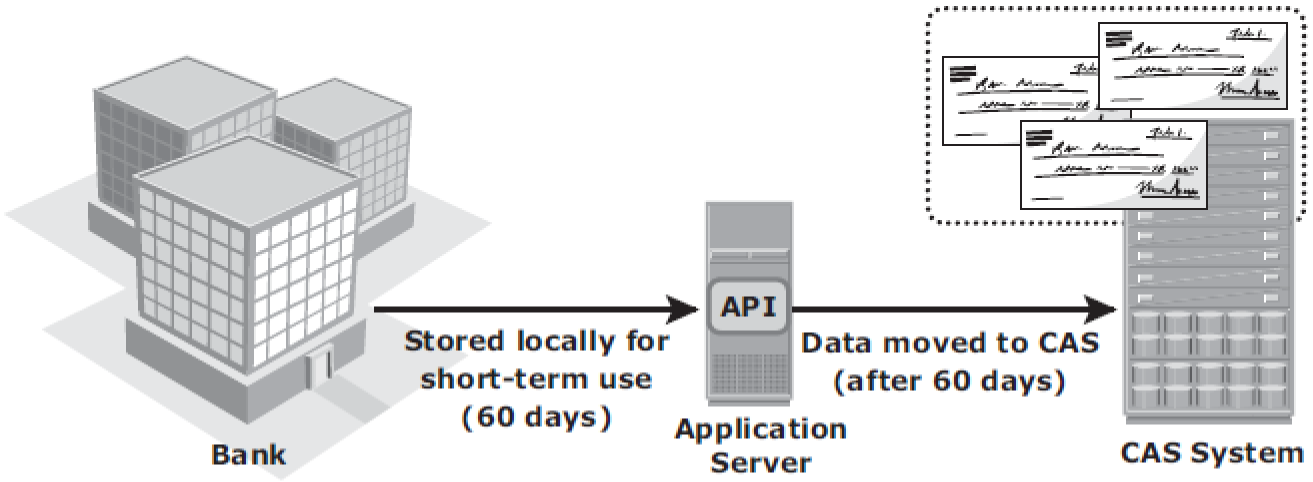


Figure 8-8: Storing financial records on a CAS system

UNIT 4- Storage Networking Technologies

Object Based and Unified Storage(Chapter 8):

Object Based Storage Devices,

Content Addressed Storage,

CAS Use Cases,

Unified Storage.

Unified Storage

Known as multiprotocol storage

Is a single data storage system that supports both file and block access.

The subsystem can support NAS (file-based access), as well as iSCSI and Fibre Channel (block-based access) storage protocols simultaneously.

RAMAIAH
Institute of Technology

Unified Storage

Components of Unified Storage

Unified storage consolidates block, file, and object access into one storage solution.

It supports multiple protocols, such as CIFS, NFS, iSCSI, FC, FCoE, REST (representational state transfer), and SOAP (simple object access protocol).

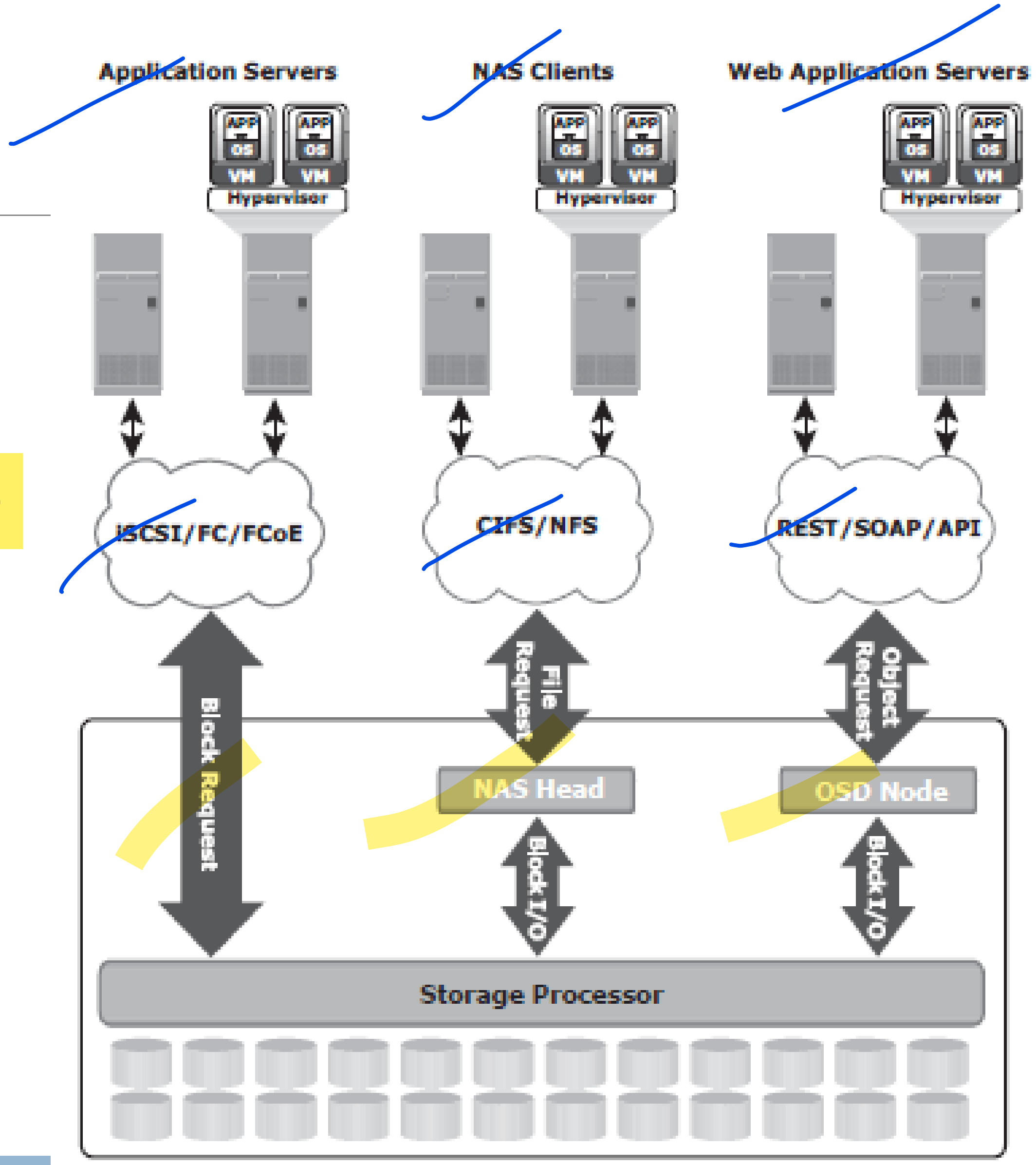
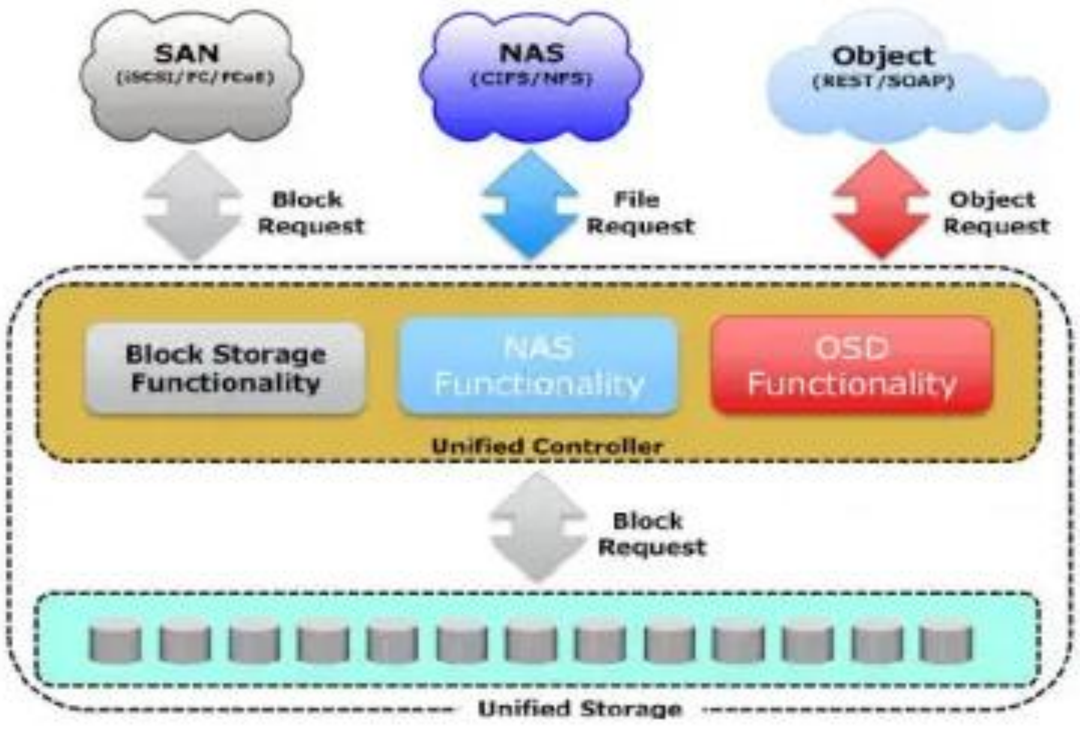


Figure 8-9: Unified storage platform

Unified Storage

Components of Unified Storage



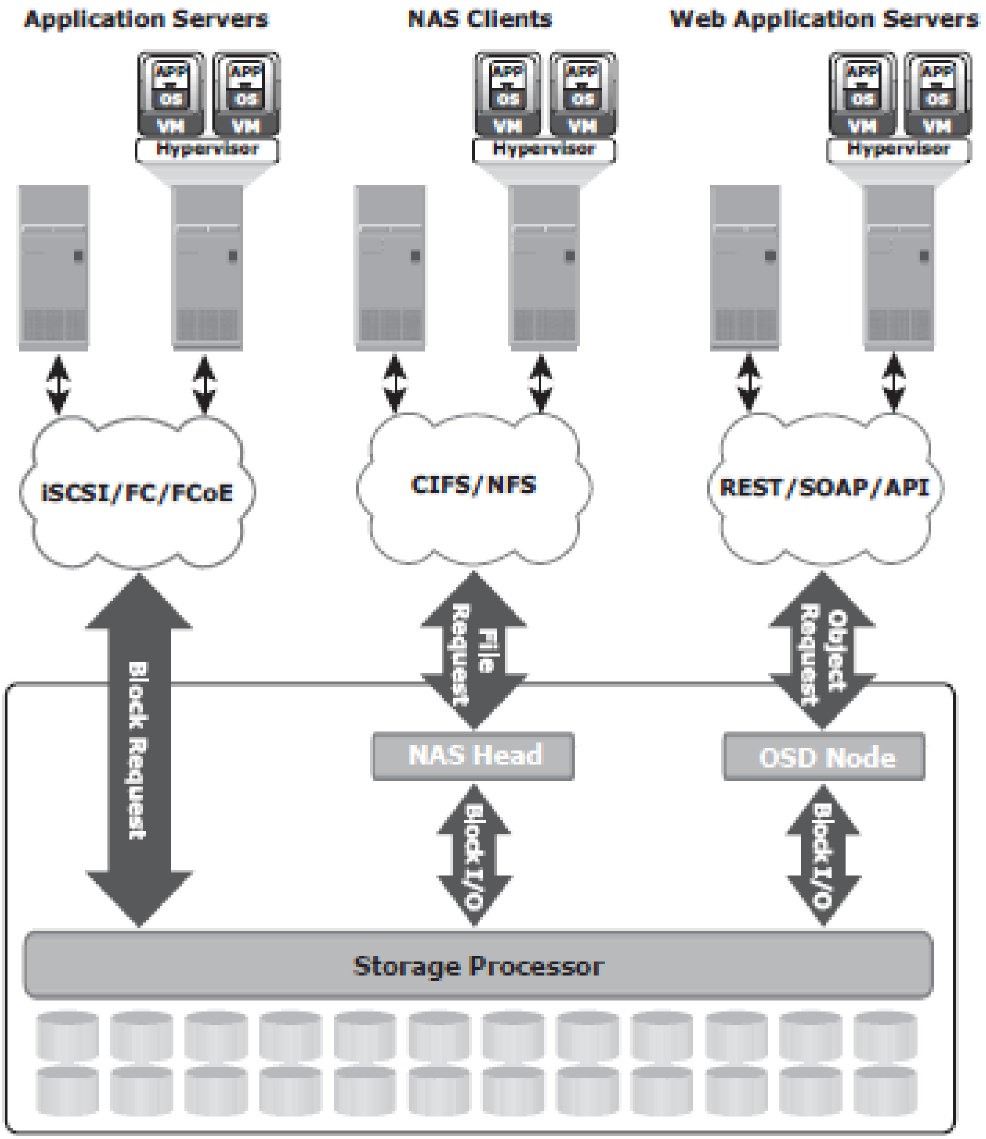


Figure 8-9: Unified storage platform

Unified Storage

Components of Unified Storage

Figure 8-9 illustrates the block diagram of a unified storage platform.

A unified storage system consists of the following key components:

1. Storage controller,
2. NAS head,
3. OSD node,
4. storage.

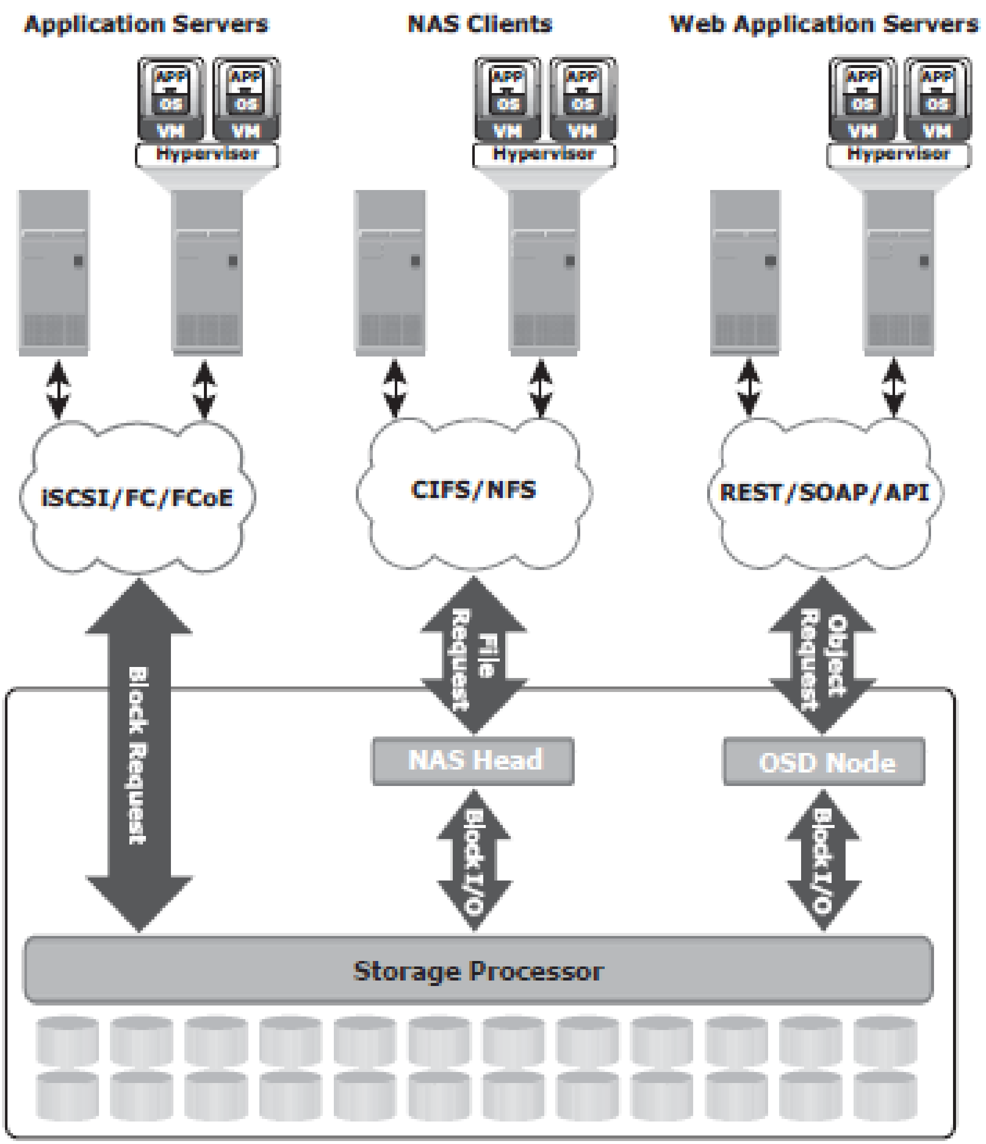


Figure 8-9: Unified storage platform

RAMAIAH
Institute of Technology

Unified Storage

Components of Unified Storage

The storage controller provides block-level access to application servers through iSCSI, FC, or FCoE protocols.

It contains iSCSI, FC, and FCoE front-end ports for direct block access.

The storage controller is also responsible for managing the back-end storage pool in the storage system.

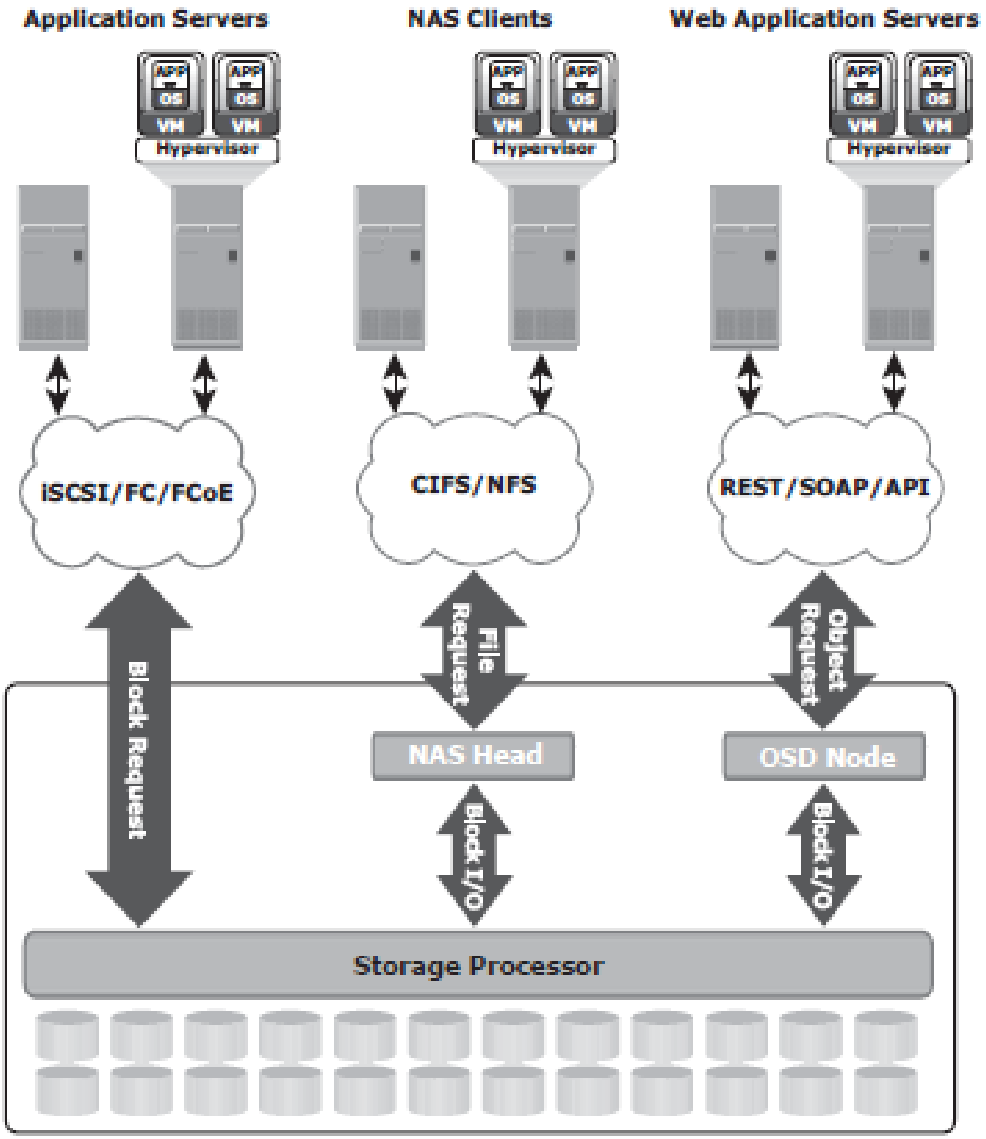


Figure 8-9: Unified storage platform

Unified Storage

Components of Unified Storage

NAS head is a dedicated file server that provides file access to NAS clients.

The NAS head is connected to the storage via the storage controller typically using a FC or FCoE connection.

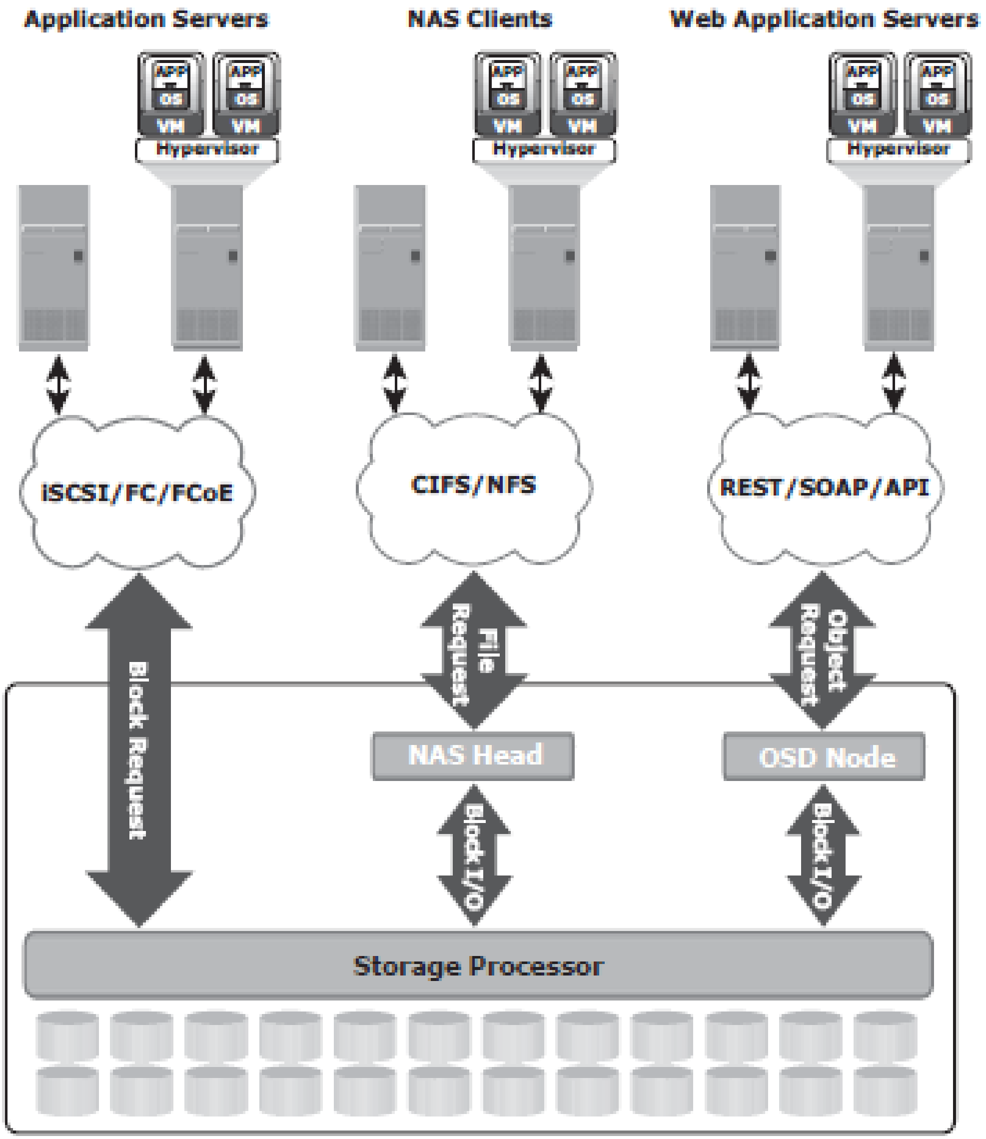


Figure 8-9: Unified storage platform

Unified Storage

Components of Unified Storage

The OSD node accesses the storage through the storage controller using a FC or FCoE connection.

The LUNs assigned to the OSD node appear as physical disks.

These disks are configured by the OSD nodes, enabling them to store the data from the web application servers.

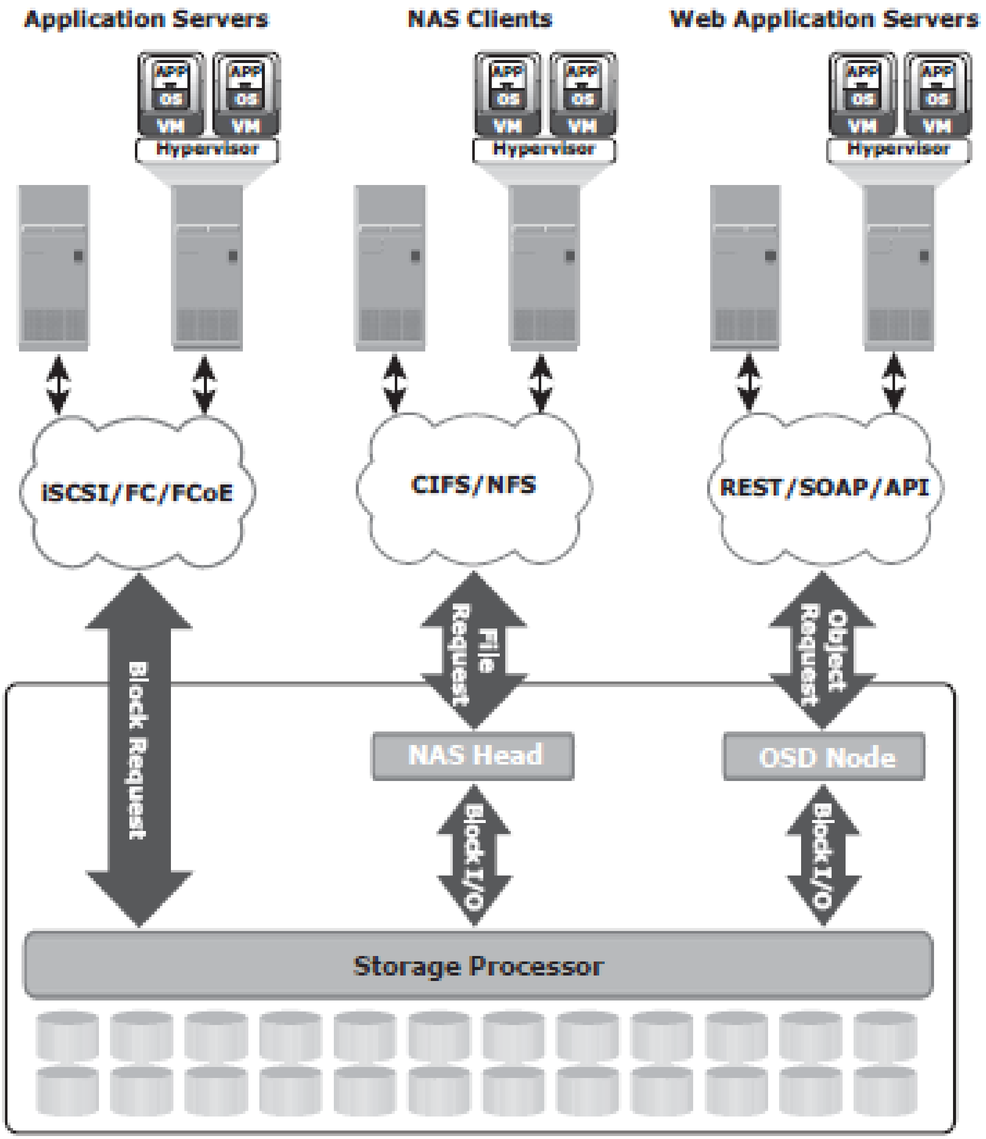


Figure 8-9: Unified storage platform

RAMAIAH
Institute of Technology

Unified Storage

Data Access from Unified Storage

Figure 8-9 illustrates the different I/O paths for block, file, and object access.

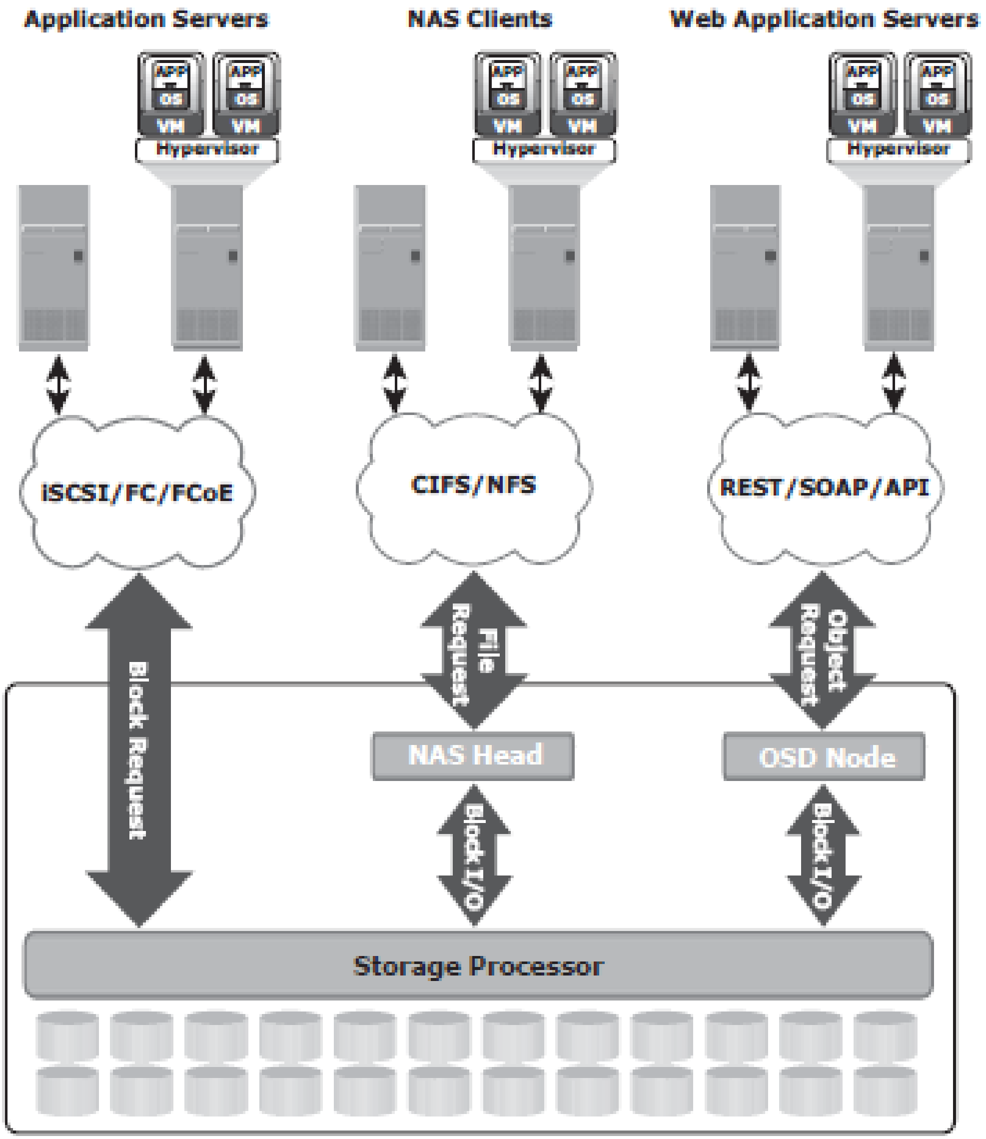


Figure 8-9: Unified storage platform

RAMAIAH
Institute of Technology

Unified Storage

Data Access from Unified Storage

Block I/O request: The application servers are connected to an FC, iSCSI,or FCoE port on the storage controller.

File I/O request: The NAS clients (where the NAS share is mounted or mapped) send a fi le request to the NAS head using the NFS or CIFS protocol.

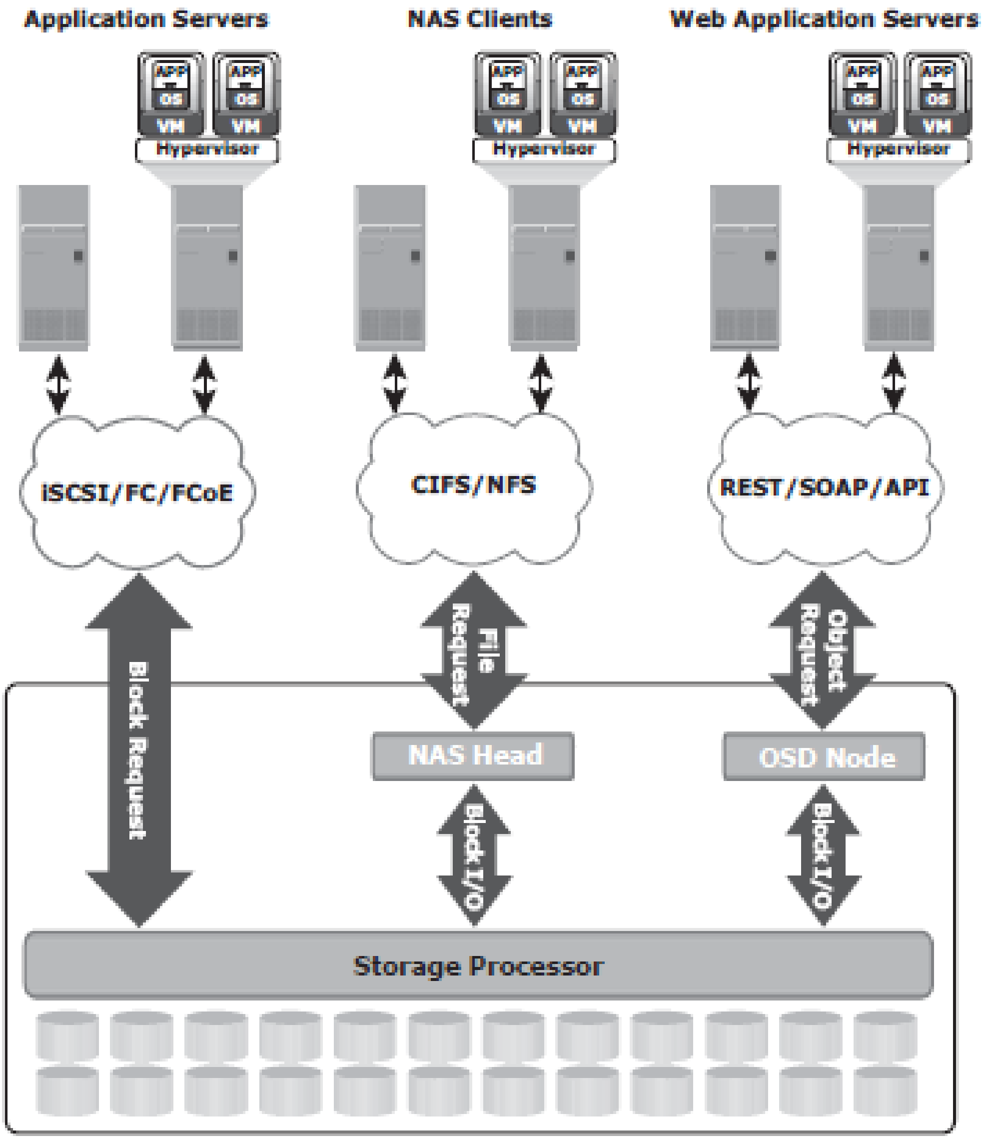


Figure 8-9: Unified storage platform

Unified Storage

Data Access from Unified Storage

Object I/O request:

The web application servers send an object request, typically using REST or SOAP protocols, to the OSD node.

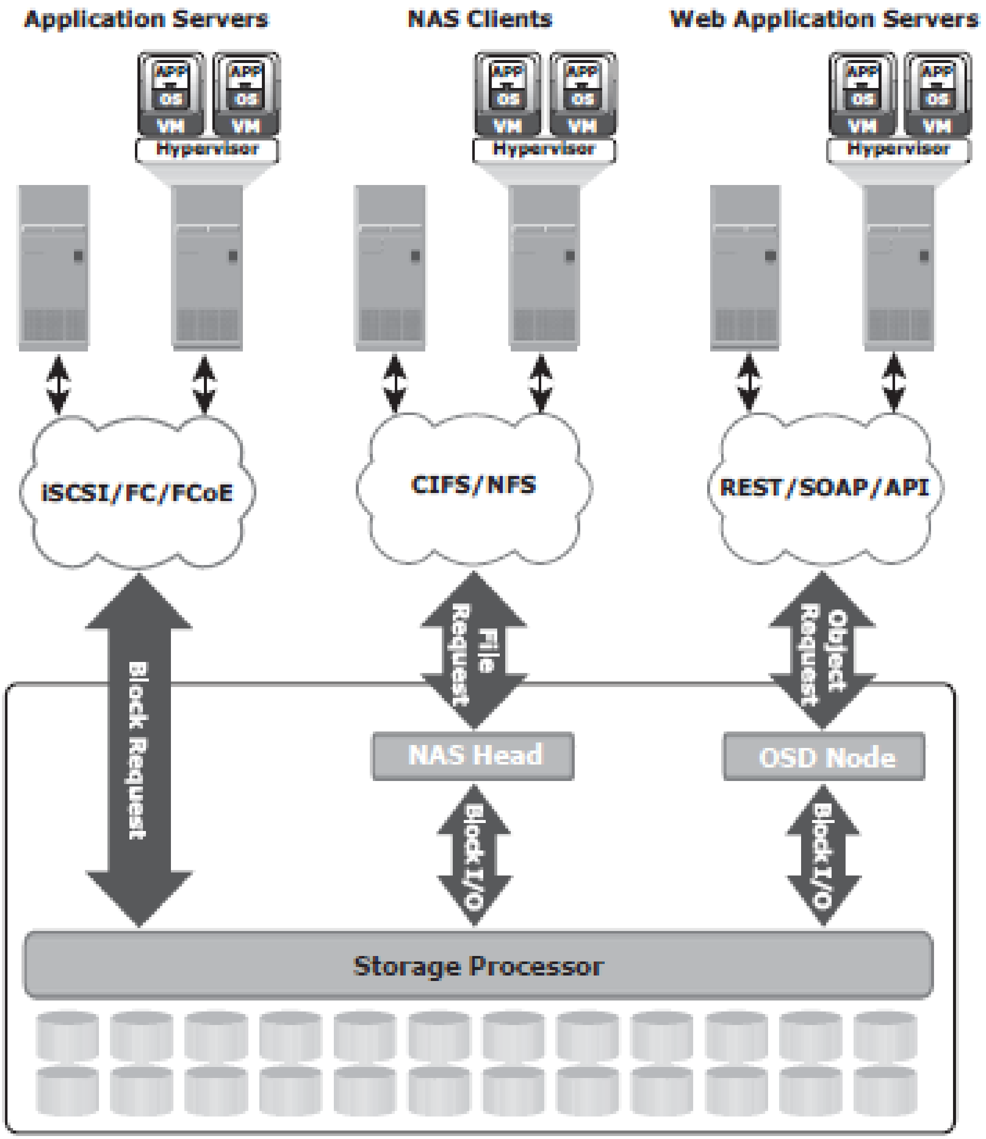


Figure 8-9: Unified storage platform

UNIT 4- Storage Networking Technologies

Network-Attached Storage(Chapter 7): Benefits of NAS, Components of NAS, NAS I/O Operation, NAS Implementations, NAS File Sharing Protocols, Factors Affecting NAS Performance, File-Level Virtualization.

Object Based and Unified Storage(Chapter 8): Object Based Storage Devices, Content Addressed Storage, CAS Use Cases, Unified Storage.


Thank
You