FUTURE VISION BIE

One Stop for All Study Materials

& Lab Programs

Future Vision

By K B Hemanth Raj

Scan the QR Code to Visit the Web Page

Or

Visit : https://hemanthrajhemu.github.io

Gain Access to All Study Materials according to VTU,
CSE – Computer Science Engineering,
ISE – Information Science Engineering,
ECE - Electronics and Communication Engineering
& MORE...

Join Telegram to get Instant Updates: https://bit.ly/VTU_TELEGRAM

Contact: MAIL: futurevisionbie@gmail.com

INSTAGRAM: www.instagram.com/hemanthraj_hemu/

INSTAGRAM: www.instagram.com/futurevisionbie/

WHATSAPP SHARE: https://bit.ly/FVBIESHARE

BMS
Institute of Technology and Management

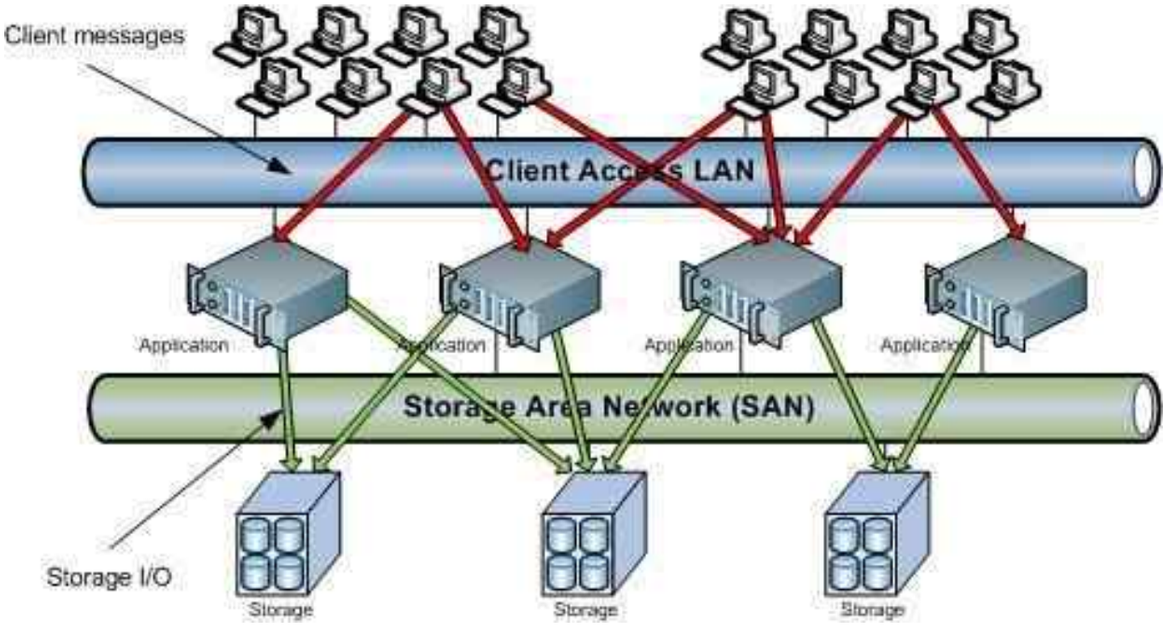
Avalahalli, Doddaballapur Main Road, Bengaluru – 560064

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Storage Area Networks

(17CS754)

SANs are primarily used to access storage devices, such as disk arrays and tape libraries from servers so that the devices appear to the operating system as direct- attached storage.



STORAGE AREA NETWORKS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 - 2018) SEMESTER – VII

Subject Code	17CS754	IA Marks	40
Number of Lecture Hours/Week	3	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Module – 1	**Teaching Hours**
Storage System Introduction to evolution of storage architecture, key data centre Elements, virtualization, and cloud computing. Key data centre elements – Host (or compute), connectivity, storage, and application in both classic and virtual Environments. RAID implementations, techniques, and levels along with the Impact of RAID on application performance. Components of intelligent storage systems and virtual storage provisioning and intelligent storage system Implementations.	**8 Hours**
Module – 2	
Storage Networking Technologies and Virtualization Fibre Channel SAN components, connectivity options, and topologies including access protection mechanism „zoning", FC protocol stack, addressing and operations, SAN-based virtualization and VSAN technology, iSCSI and FCIP(Fibre Channel over IP) protocols for storage access over IP network, Converged protocol FCoE and its components, Network Attached Storage (NAS) - components, protocol and operations, File level storage virtualization, Object based storage and unified storage platform.	**8 Hours**
Module – 3	
Backup, Archive, and Replication This unit focuses on information availability and business continuity solutions in both virtualized and non-virtualized environments. Business continuity terminologies, planning and solutions, Clustering and multipathing architecture to avoid single points of failure, Backup and recovery - methods, targets and topologies, Data deduplication and backup in virtualized environment, Fixed content and data archive, Local replication in classic and virtual environments, Remote replication in classic and virtual environments, Three-site remote replication and continuous data protection	**8 Hours**
Module – 4	
Cloud Computing Characteristics and benefits This unit focuses on the business drivers, definition, essential characteristics, and phases of journey to the Cloud. ‚Business drivers for Cloud computing, Definition of Cloud computing, Characteristics of Cloud computing, Steps involved in transitioning from Classic data center to Cloud computing environment Services and deployment models, Cloud infrastructure components, Cloud migration considerations	**8 Hours**
Module – 5	
Securing and Managing Storage Infrastructure This chapter focuses on framework and domains of storage security along with covering security. implementation at storage networking. Security threats, and countermeasures in various domains Security solutions for (Fiber channel)FC-SAN, IP-SAN, and NAS	**8 Hours**

managing various information infrastructure components in classic and virtual environments, Information lifecycle management (ILM) and storage tiering, Cloud service management activities	

Course outcomes: The students should be able to:

- Identify key challenges in managing information and analyze different storage networking technologies and virtualization
- Explain components and the implementation of NAS
- Describe CAS architecture and types of archives and forms of virtualization
- Illustrate the storage infrastructure and management activities

Question paper pattern:

The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Information Storage and Management, Author :EMC Education Services, Publisher: Wiley ISBN: 9781118094839
2. Storage Virtualization, Author: Clark Tom, Publisher: Addison Wesley Publishing Company ISBN: 9780321262516

Table of Content

Sl.No	Module	Page No.
1	Module – 1	5
2	Module – 2	27
3	Module – 3	120
4	Module – 4	220
5	Module – 5	236

Module-2

Storage Networking Technologies and Virtualization

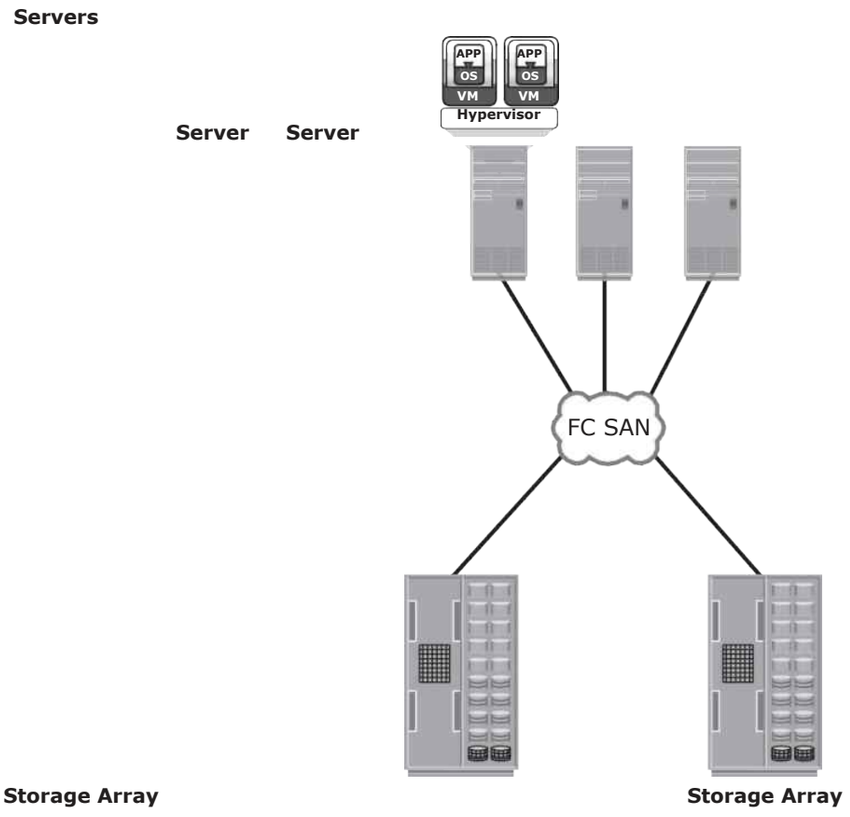
Fibre Channel: Overview

The FC architecture formsthe fundamentalconstruct of the FC SANinfrastructure. *Fibre Channel* is a high-speed network technology that runs on high-speed optical fibercablesand serial coppercables. The FC technology was developed to meetthe demand for increased speeds of data transfer between servers and mass storage systems. Although FC networking was introduced in 1988, the FC standardization process began when the American National Standards Institute (ANSI) chartered the Fibre Channel Working Group(FCWG). By 1994, the new high-speed computer interconnection standard was developedand the Fibre Channel Association(FCA) was founded with 70 charter member companies. Technical Committee T11, which is the committee within International Committee for Information Technology Standards (INCITS), is responsible for Fibre Channel interfacestandards.

High data transmission speed is an important feature of the FC networking technology. The initial implementation offered a throughput of 200 MB/s (equivalent to a raw bit rate of 1Gb/s), which was greater than the speeds of Ultra SCSI (20 MB/s), commonly used in DAS environments. In comparison with Ultra SCSI, FC is a significant leap in storage networking technology. The latest FC implementations of 16 GFC (Fibre Channel) offer a throughput of 3200 MB/s (raw bit rates of 16 Gb/s), whereas Ultra640 SCSI is available with a throughput of 640 MB/s. The FC architecture is highly scalable, and theoreti- cally, a single FC network can accommodate approximately 15 million devices.

The SAN and Its Evolution

A SAN carries data between servers (or *hosts*) and storage devices through Fibre Channel network (see - 5-1). A SAN enables storage consolidation and enables storage to be shared across multiple servers. This improves the utilization of storage resources compared to direct-attached storage architecture and reduces the total amount of storage an organization needs to purchase and manage. With consolidation, storage management becomes centralized and less complex, which further reduces the cost of managing information. SAN also enables organizations to connect geographically dispersed servers and storage.

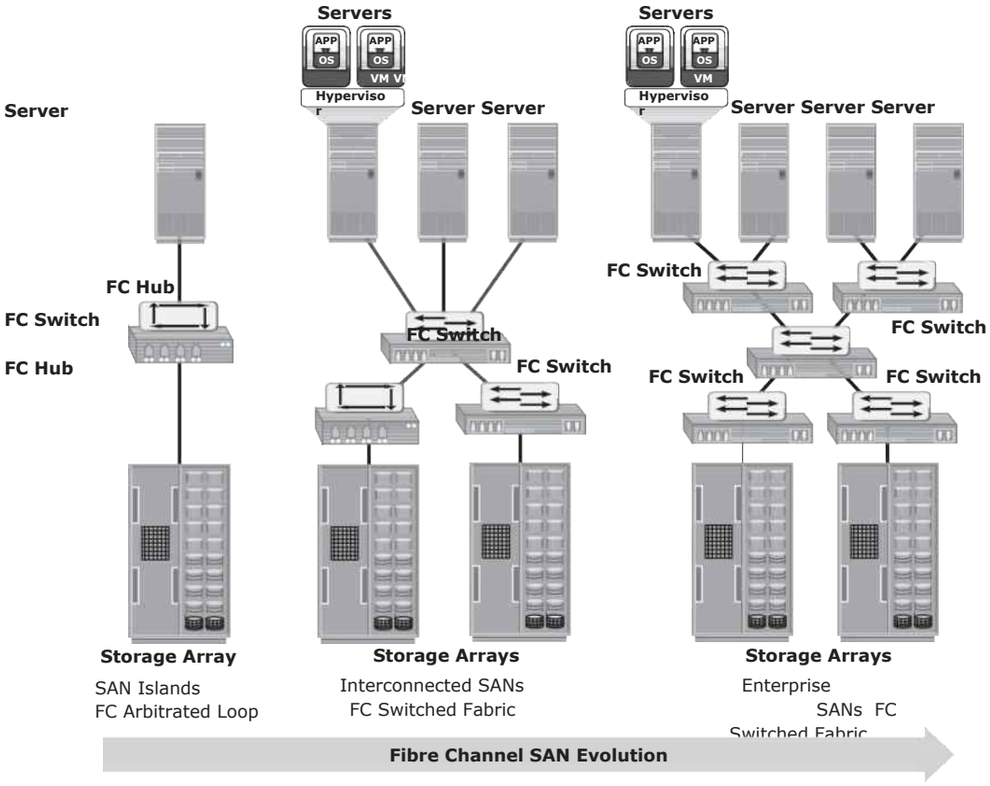


- 5-1: FC SAN implementation

In its earliest implementation, the FC SAN was a simple grouping of hosts and storage devices connected to a network using an FC hub as a connectivity device. This configuration of an FC SAN is known as a *Fibre Channel Arbitrated*

Loop (FC-AL). Use of hubs resulted in isolated FC-AL SAN islands because hubs provide limited connectivity and bandwidth.

The inherent limitations associated with hubs gave way to high-performance FC *switches*. Use of switches in SAN improved connectivity and performance and enabled FC SANs to be highlyscalable. This enhanced dataaccessibility to applica- tions across the enterprise. Now, FC-AL has been almost abandoned for FC SANs due to its limitations but still survives as a back-end connectivity option todisk drives. - 5-2 illustratesthe FC SANevolutionfrom FC-AL to enterprise SANs.



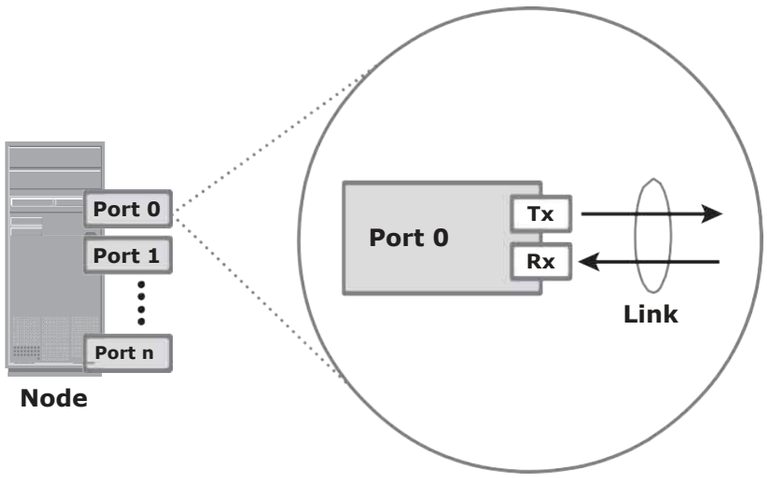
- 5-2: FC SAN evolution

Components of FC SAN

FC SAN is a network of servers and shared storage devices. Servers and storage are the end points or devices in the SAN (called *nodes*). FC SAN infrastructure consists of node ports, cables, connectors, and interconnecting devices (such as FC switches or hubs), along with SAN management software.

Node Ports

In a Fibre Channel network, the end devices, such as hosts, storage arrays, and tape libraries, are all referred to as *nodes*. Each node is a source or destination of information. Each node requires one or more ports to providea physicalinterface for communicating with other nodes. These ports are integral components of host adapters, such as HBA, and storage front-end controllers or adapters. In an FC environment a port operates in full-duplex data transmission mode with a *transmit* (Tx) link and a *receive* (Rx) link (see - 5-3).

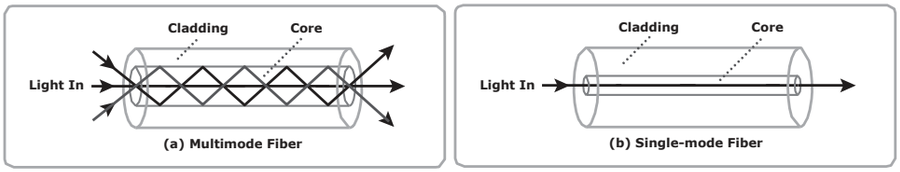


- 5-3: Nodes, ports, and links

Cables and Connectors

SAN implementations use optical fiber cabling. Copper can be used for shorter distances for back-end connectivity because it provides an acceptable signal-to- noise ratio for distances up to 30 meters. Optical fiber cables carry data in the form of light. There are two types of optical cables: multimode and single-mode. *Multimode fiber* (MMF) cable carries multiple beams of light projected at differ- ent angles simultaneously onto the core of the cable (see - 5-4 [a]). Based on the bandwidth, multimode fibers are classified as OM1 (62.5µmcore), OM2 (50µm core), and laser-optimized OM3 (50µm core). In an MMF transmission, multiple light beams traveling inside the cable tend to disperse and collide. This collision weakens the signal strength after it travels a certain distance — a process known as *modal dispersion*. An MMF cable is typically used for short distances because of signal degradation (attenuation) due to modal dispersion. *Single-mode fiber* (SMF) carries a single ray of light projected at the center of the core (see - 5-4 [b]). These cables are available in core diameters of 7 to 11microns;

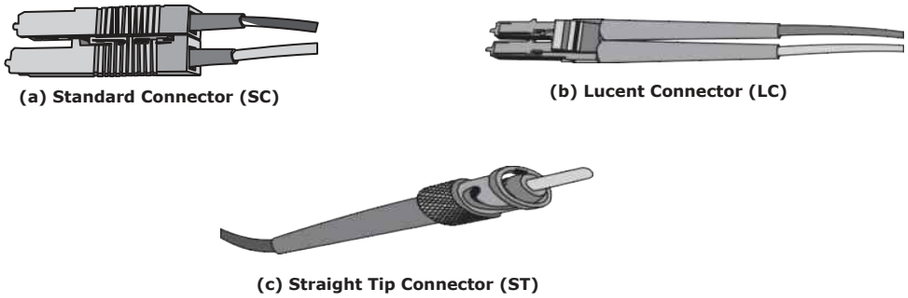
the most common size is 9 microns. In an SMF transmission, a single light beam travels in a straight line through the core of the fiber. The small core and the single light wave help to limit modal dispersion. Among all types of fiber cables, single- mode provides minimum signal attenuation over maximum distance (up to 10 km). A single-mode cable is used for long-distance cable runs, and distance usually depends on the power of the laser at the transmitter and sensitivity of the receiver.



- 5-4: Multimode fiber and single-mode fiber

MMFs are generally used withindatacentersforshorterdistance runs, whereas SMFs are used for longer distances.

A connector is attached at the end of a cable to enable swift connection and disconnection of the cable to and from a port. A *Standard connector* (SC) (see - 5-5 [a]) and a *Lucent connector* (LC) (see - 5-5 [b]) are two commonly used connectors for fiber optic cables. *Straight Tip* (ST) is another fiber-optic con- nector, which is often used with fiber patch panels (see - 5.5 [c]).



- 5-5: SC, LC, and ST connectors

Interconnect Devices

FC hubs, switches, and directors are the interconnect devices commonly used in FC SAN.

Hubs are used as communication devices in FC-AL implementations. Hubs physically connect nodes in a logical loop or a physical star topology. All the nodes must share the loop because data travels through all the connection points. Because of the availability of low-cost and high-performance switches, hubs are no longer used in FC SANs.

Switches are more intelligent than hubs and directly route data from one physical port to another. Therefore, nodes do not share the bandwidth. Instead, each node has a dedicated communication path.

Directors are high-end switches with a higher port count and better fault-tolerance capabilities.

Switches are available with a fixed port count or with modular design. In a modular switch, the port count is increased by installing additional port cards to open slots. The architecture of a director is always modular, and its port count is increased by inserting additional line cards or blades to the director's chassis. High-end switches and directors contain redundant components to provide high availability. Both switches and directors have management ports (Ethernet or serial) for connectivity to SAN management servers.

A port card or blade has multiple ports for connecting nodes and other FC switches. Typically, a Fibre Channel transceiver is installed at each port slot that houses the transmit (Tx) and receive (Rx) link. In a transceiver, the Tx and Rx links share common circuitry. Transceivers inside a port card are connected to an application specific integrated circuit, also called port ASIC. Blades in a director usually have more than one ASIC for higher throughput.

SAN Management Software

SAN management software manages the interfaces between hosts, interconnect devices, and storage arrays. The software provides a view of the SAN environment and enables management of various resources from one central console.

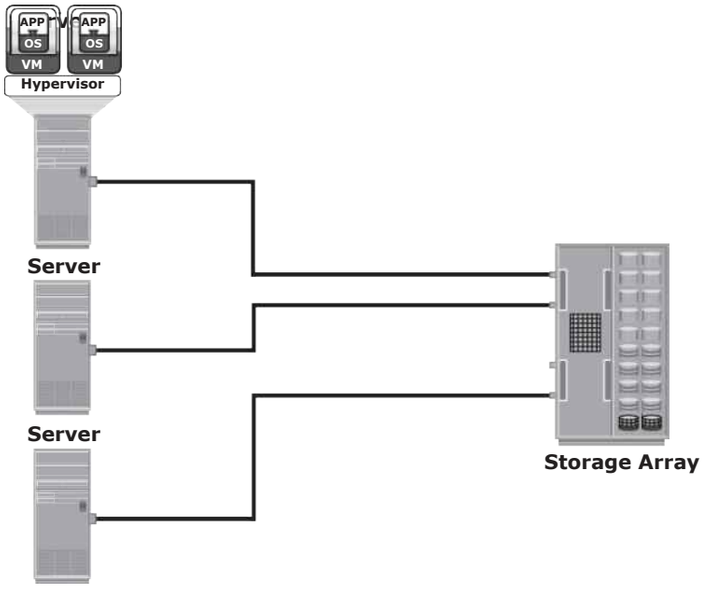
It provides key management functions, including mapping of storage devices, switches, and servers, monitoring and generating alerts for discovered devices, and *zoning* (discussed in section 5.9 —Zoning‖ later in this chapter).

FC Connectivity

The FC architecture supports three basic interconnectivity options: point-to-point, arbitrated loop, and Fibre Channel switched fabric.

5.1.1 Point-to-Point

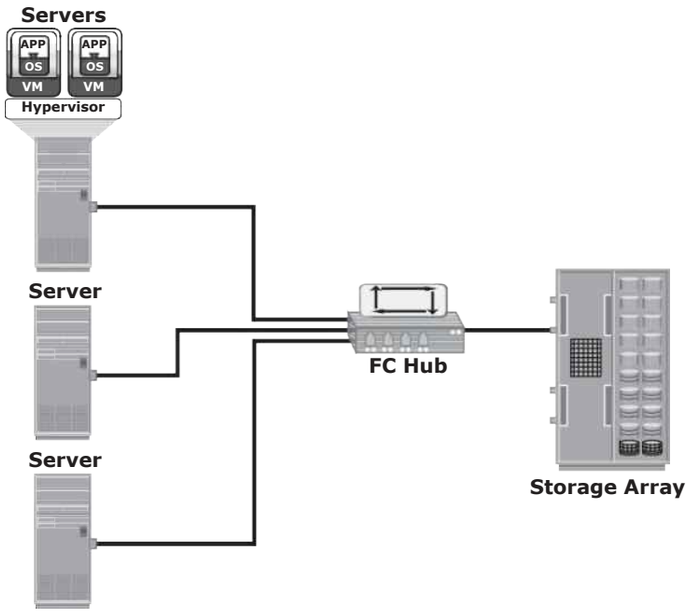
Point-to-point is the simplest FC configuration— two device s are connected directly to each other, as shown in - 5-6. This configuration provides a dedicated connection for data transmissionbetweennodes. However, the point-to-point con- figuration offers limited connectivity, because only two devices can communicate with each other at a given time. Moreover, it cannot be scaled to accommodate a large number of nodes. Standard DAS uses point-to-point connectivity.



- 5-6: Point-to-point connectivity

Fibre Channel Arbitrated Loop

In the FC-AL configuration, devices are attached to a shared loop. FC-AL has the characteristics of a token ring topology and a physical star topology. In FC-AL, each device contends with other devices to perform I/O operations. Devices on the loop must —arbitratell to gain control of the loop. At any given time, only one device can perform I/O operations on the loop (see - 5-7).



- 5-7: Fibre Channel Arbitrated Loop

As a loop configuration, FC-AL can be implemented without any interconnecting devices by directly connecting one device to another two devices in a ring through cables.

However, FC-AL implementations may also use hubs whereby the arbitrated loop is physically connected in a star topology.

The FC-AL configuration has the following limitations in terms of scalability:

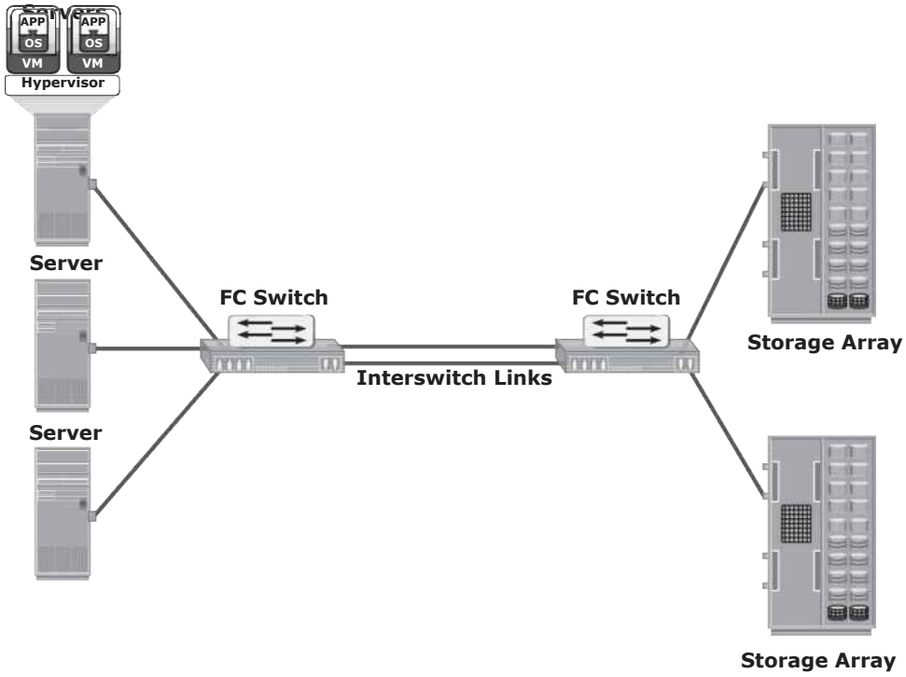
- FC-AL shares the loop and only one device can perform I/O operations at a time. Because each device in a loop must wait for its turn to process an I/Orequest, theoverallperformancein FC-ALenvironmentsislow.
- FC-AL uses only 8-bits of 24-bit Fibre Channel addressing (the remaining 16-bits are masked) and enables the assignment of 127 valid addresses to the ports. Hence, it can support up to 127 devices on a loop. One address is reserved foroptionally connectingtheloopto anFC switchport. Therefore, up to 126 nodes can be connected to the loop.
- Adding or removing a device results in loop re-initialization, which can cause a momentary pause in looptraffic.

Fibre Channel Switched Fabric

Unlike a loop configuration, a Fibre Channel switched fabric (FC-SW) network provides dedicated data path and scalability. The addition or removal of a device

in a switched fabric is minimally disruptive; it does not affect the ongoing traffic between other devices.

FC-SW is also referred to as *fabric connect*. A fabric is a logical space in which all nodes communicate with one another in a network. This virtual space can be created with a switch or a network of switches. Each switch in a fabric contains a unique domain identifier, which is part of the fabric's addressing scheme. In FC-SW, nodes do notshare a loop; instead, datais transferred through adedicated path between the nodes. Each port in a fabric has a unique 24-bit Fibre Channel address for communication. - 5-8 shows an example of the FC-SW fabric. In a switched fabric, the link between any two switches is called an *Interswitch link* (ISL). ISLs enable switches to be connected together to form a single, larger fabric. ISLs are used totransfer host-to-storage data and fabricmanagementtraffic from one switch to another. By using ISLs, a switched fabric can be expanded to connect a large number of nodes.

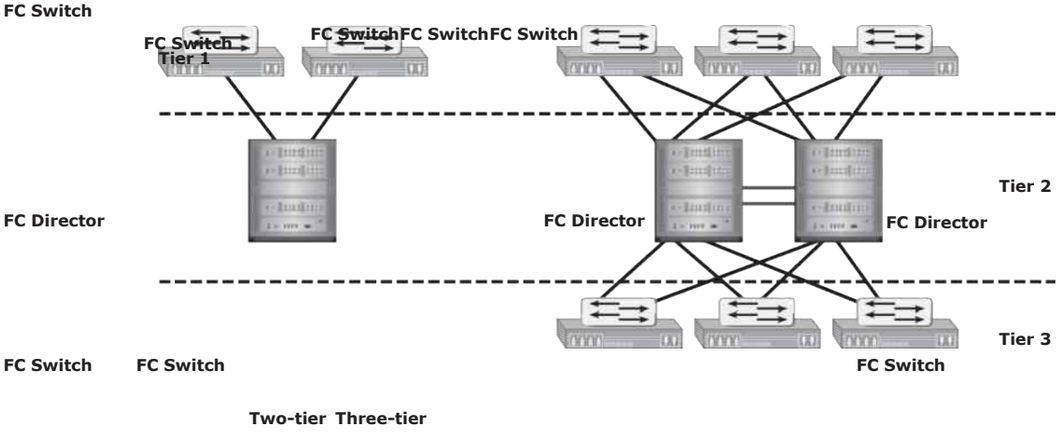


- 5-8: Fibre Channel switched fabric

A fabric can be described by the number of tiers it contains. The number of tiers in a fabric is based on the number of switches traversed between two points that are farthest from each other. This number is based on the infrastructure

constructed by the fabric instead of how the storage and server are connected across the switches.

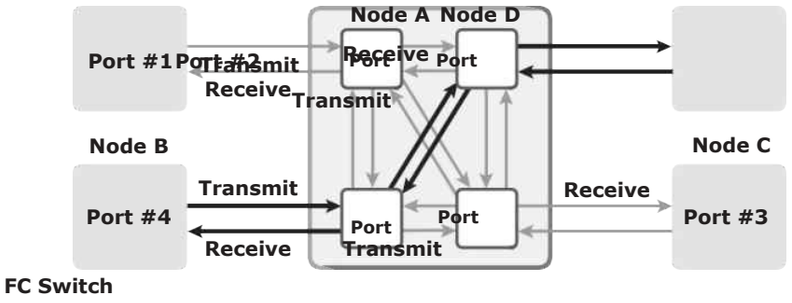
When the number of tiers in a fabric increases, the distance that the fabric management traffic must travelto reach each switch also increases. This increase in the distance also increases the time taken to propagate and complete a fabric reconfiguration event, such as the addition of a new switch or a zone set propa- gation event. - 5-9 illustrates two-tier and three-tier fabric architecture.



- 5-9: Tiered structure of Fibre Channel switched fabric

FC-SW Transmission

FC-SW uses switches that can switch datatraffic between nodes directly through switch ports. Frames are routed between source and destination bythe fabric. As shown in - 5-10, if node B wants to communicate with node D, the nodes should individually login first and then transmit data via the FC-SW. This link is considered a dedicated connection between the initiator and the target.



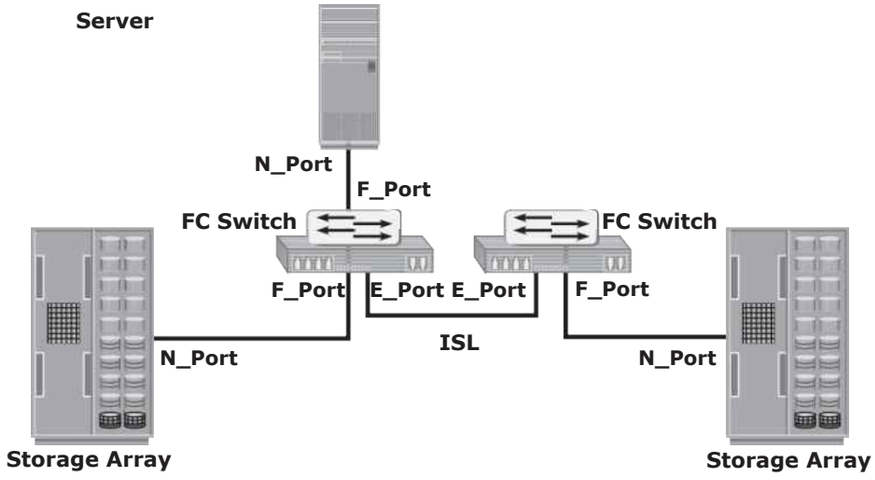
- 5-10: Data transmission in Fibre Channel switched fabric

Switched Fabric Ports

Ports in a switched fabric can be one of the following types:

- **N_Port:** An end point in the fabric. This port is also known as the *node port*. Typically, it is a host port (HBA) or a storage array port connected to a switch in a switched fabric.
- **E_Port:** A port that forms the connection between two FC switches. This portis also knownasthe *expansionport*. The E_Portonan FC switchconnects to the E_Port of another FC switch in the fabric through ISLs.
- **F_Port:** A port on a switch that connects an N_Port. It is also known as a *fabric port*.
- **G_Port:** A generic port on a switch that can operate as an E_Port or an F_Port and determines its functionality automatically duringin itialization.

- 5-11 shows various FC ports located in a switched fabric.



- 5-11: Switched fabric ports

Fibre Channel Architecture

Traditionally, hostcomputeroperatingsystems havecommunicated withperipheral devices over channel connections, such as ESCON and SCSI. Channel technologies provide high levels of performance with low protocol overheads. Such performance is achievable due to the static nature of channels and the high level of hardware and software integration provided by the channel technologies.

However, these technologies suffer from inherent limitations in terms of the number of devices that can be connected and the distance between these devices. In contrast to channel technology, network technologies are more flexible and provide greater distance capabilities. Network connectivity provides greater scalabilityandusessharedbandwidthforcommunication.Thisflexibilityresults in greater protocol overhead and reduced performance.

The FC architecture representstrue channel/networkintegration and captures some of the benefits of both channel and network technology. FC SAN uses the

Fibre Channel Protocol (FCP) that provides both channel speed for data transfer

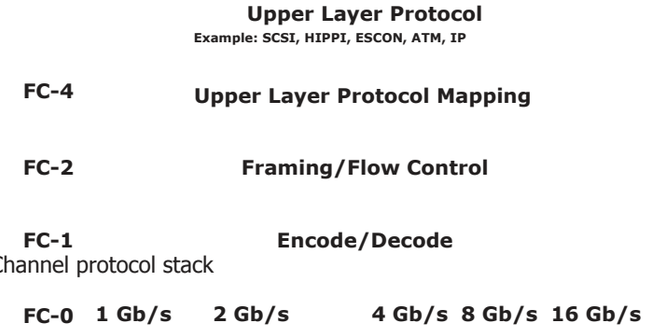
with low protocol overhead and scalability of network technology.

FCPformsthefundamentalconstructoftheFCSANinfrastructure.FibreChannel provides a serial data transfer interface that operates over copper wire and optical fiber. FCP is the implementation of serial SCSI over an FC network. In FCP archi- tecture, all external and remote storage devicesattached to the SANappear as local devices to the host operating system. The key advantages of FCP are as follows:

- Sustained transmission bandwidth over long distances.
- Support for a larger number of addressable devices over a network. Theoretically, FCcansupportmorethan 15 million device addresseson a network.
- Support speeds up to 16 Gbps (16 GFC).

Fibre Channel Protocol Stack

It is easier to understand a communication protocol by viewing it as a structure of independent layers. FCP defines the communication protocol in five layers: FC-0 through FC-4 (except FC-3 layer, which is not implemented). In a layered communication model, the peer layers on each node talk to each other through defined protocols. - 5-12 illustrates the Fibre Channel protocol stack.



- 5-12: Fibre Channel protocol stack

FC-4 Layer

FC-4 is the uppermost layer in the FCP stack. This layer defines the application interfaces and the way *Upper Layer Protocols* (ULPs) are mapped to the lower FC layers. The FC standard defines several protocols that can operate on the FC-4 layer (see - 5-12). Some of the protocols include SCSI, High Performance Parallel Interface (HIPPI) Framing Protocol, Enterprise Storage Connectivity (ESCON), Asynchronous Transfer Mode (ATM), and IP.

FC-2 Layer

The FC-2 layer provides Fibre Channel addressing, structure, and organization of data (frames, sequences, and exchanges). It also defines fabric services, classes of service, flow control, and routing.

FC-1 Layer

The FC-1 layer defines how data is encoded prior to transmission and decoded upon receipt. At the transmitter node, an 8-bit character is encoded into a 10-bit transmissions character. This character is then transmitted to the receiver node. At the receiver node, the 10-bit character is passed to the FC-1 layer, which decodes the 10-bit character into the original 8-bit character. FC links with speeds of 10 Gbps and above use 64-bit to 66-bit encoding algorithms. The FC-1 layer also defines the transmission words, such as FC frame delimiters, which identify the start and end of a frame and primitive signals that indicate events at a transmitting port. In addition to these, the FC-1 layer performs link initialization and error recovery.

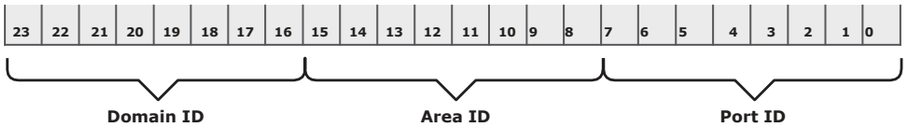
FC-0 Layer

FC-0 is the lowest layer in the FCP stack. This layer defines the physical interface, media, and transmission of bits. The FC-0 specification includes cables, connec- tors, and optical and electrical parameters for a variety of data rates. The FC transmission can use both electrical and optical media.

Mainframe SANs use *Fibre Connectivity* (FICON) for a low-latency, high-bandwidth connection to the storage controller. FICON was designed as a replacement for *Enterprise System Connection* (ESCON) to support mainframe-attached storage systems.

Fibre Channel Addressing

An FC address is dynamically assigned when a node port logs on to the fabric. The FC address has a distinct format, as shown in - 5-13. The address- ing mechanism provided here corresponds to the fabric with the switch as an interconnecting device.



- 5-13: 24-bit FC address of N_Port

The first field of the FC address contains the domain ID of the switch. A *domain ID* is a unique number provided to each switch in the fabric. Although this is an 8-bit field, there are only 239 available addresses for domain ID because some addresses are deemed special and reserved for fabric management services. For example, FFFFFC is reserved for the name server, and FFFFFE is reserved for the fabric login service. The *area ID* is used to identify a group of switch ports used for connecting nodes. An example of a group of ports witha commonarea ID is a port card on the switch. The last field, the *port ID*, identifies the port within the group.

Therefore, the maximum possible number of node ports in a switched fabric is calculated as:

239 domains ∞ 256 areas ∞ 256 ports = 15,663,104

World Wide Names

Each device in the FC environment is assigned a 64-bit unique identifier called the *World Wide Name* (WWN). The Fibre Channel environment uses two types of WWNs: *World Wide Node Name* (WWNN) and *World Wide Port Name* (WWPN). Unlike an FC address, which is assigned dynamically, a WWN is a static name

for each node on an FC network. WWNs are similar to the Media Access Control (MAC) addresses used in IP networking. WWNs are *burned* into the hardware or assigned through software. Several configuration definitions in a SAN use WWN for identifying storage devices and HBAs. The name server in an FC environment keeps the association of WWNs to the dynamically created FC addresses for nodes. - 5-14 illustrates the WWN structure examples for an array and an HBA.

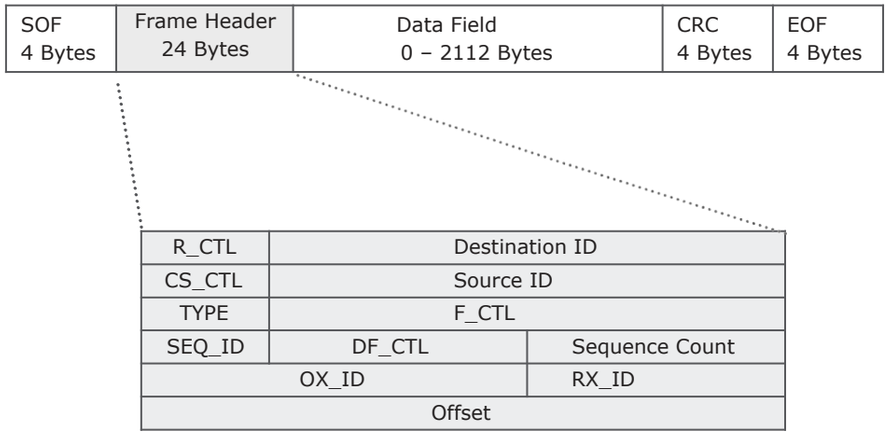
World Wide Name - Array															
5	0	0	6	0	1	6	0	0	0	6	0	0	1	B	2
0101	0000	0000	0110	0000	0001	0110	0000	0000	0000	0110	0000	0000	0001	1011	0010
Format Type	Company ID 24 bits						Port	Model Seed 32 bits							

World Wide Name - HBA															
1	0	0	0	0	0	0	0	c	9	2	0	d	c	4	0
Format Type	Reserved 12 bits			Company ID 24 bits						Company Specific 24 bits					

- 5-14: World Wide Names

5.1.2 FC Frame

An FC frame consists of five parts: *start of frame* (SOF), *frame header*, *data field*, *cyclic redundancy check* (CRC), and *end of frame* (EOF).



- 5-15: FC frame

The SOF and EOF act as delimiters. In addition to this role, the SOF also indicates whether the frame is the first frame in a sequence of frames.

The frame header is 24 bytes long and contains addressing information for the frame. It includes the following information: Source ID (S_ID), Destination ID (D_ID), Sequence ID (SEQ_ID), Sequence Count (SEQ_CNT), Originating Exchange ID (OX_ID), and Responder Exchange ID (RX_ID), in addition to some control fields.

The S_ID and D_ID are FC addresses for the source port and the destination port, respectively. The SEQ_ID and OX_ID identify the frame as a component of a specific sequence and exchange, respectively.

The frame header also defines the following fields:

- **Routing Control (R_CTL):** This field denotes whether the frame is alink control frame oradata frame. Linkcontrolframesareframesthatdonot carry any user data. These frames are used for setup and messaging. In contrast, data frames carry the user data.
- **Class Specific Control (CS_CTL):** This field specifies link speeds for class 1 and class 4 data transmission. (Class of service is discussed in section 5.6.7 —Classes of Service‖ later in the chapter.)
- **TYPE:** This field describes the upper layer protocol (ULP) to be carried on the frame if it is a data frame. However, if it is a link control frame, this field is used to signal an event such as —fabric busy.‖ For example, if the TYPE is 08, and the frame is a data frame, it means that the SCSI will be carried on an FC.
- **Data Field Control (DF_CTL):** A 1-byte field that indicates the existence of any optional headers at the beginning of the data payload. It is a mecha- nism to extend header information into the payload.
- **Frame Control (F_CTL):** A 3-byte field that contains control information related to frame content.Forexample,one ofthebitsinthisfieldindicates whether this is the first sequence of the exchange.

The data field in an FC frame contains the data payload, up to 2,112 bytes of actual data with 36 bytes of fixed overhead.

The CRC checksum facilitates error detection for the content of the frame. This checksum verifies data integrity by checking whether the content of the frames are received correctly. The CRC checksum is calculated by the sender before encoding at the FC-1 layer. Similarly, it is calculated by the receiver after decoding at the FC-1 layer.

Structure and Organization of FC Data

In an FC network, data transport is analogous to a conversation between two people, whereby a frame represents a word, a sequence represents a sentence, and an exchange represents a conversation.

- **Exchange:** An exchange operation enables two node ports to identify and manage a set of information units. Each upper layer protocol hasits protocol-specific information that must be sent to another port to perform certainoperations. This protocol-specific information iscalled an informa- tion unit. The structure of these information units is defined in the FC-4 layer. This unit maps to a sequence. An exchange is composed of one or more sequences.
- **Sequence:** A sequence refers to a contiguous set of frames that are sent from one port to another. A sequence corresponds to an information unit, as defined by the ULP.
- **Frame:** A frame is the fundamental unit of data transfer at Layer 2. Each frame can contain up to 2,112 bytes of payload.

Flow Control

Flow control defines the pace of the flow of data frames during data transmission. FC technology uses two flow-control mechanisms: buffer-to-buffer credit (BB_Credit) and end-to-end credit (EE_Credit).

BB_Credit

FC uses the *BB_Credit* mechanism for flow control. BB_Credit controls the maxi- mum number of frames that can be present over the link at any given point in time. In a switched fabric, BB_Credit management may take place between any two FC ports. The transmitting port maintains a count of free receiver buf- fers and continues to send frames if the count is greater than 0. The BB_Credit mechanism uses *Receiver Ready* (R_RDY) primitive that indicates a buffer has been freed on the port that transmitted the R_RDY.

EE_Credit

The function of end-to-end credit, known as *EE_Credit*, is similar to that of BB_Credit. When an initiator and a target establish themselves as nodes communicating with each other, they exchange the EE_Credit parameters (part of Port login). The EE_Credit mechanism provides the flow control for class 1 and class 2 traffic only.

Classes of Service

The FC standards define different classes of service to meet the requirements of a wide range of applications. Table 5-1 shows three classes of services and their features.

Table 5-1: FC Class of Services

	CLASS 1	CLASS 2	CLASS 3
Communication type	Dedicated connection	Nondedicated connection	Nondedicated connection
Flow control	End-to-end credit	End-to-end credit B-to-B credit	B-to-B credit
Frame delivery	In order delivery	Order not guaranteed	Order not guaranteed
Frame acknowledgment	Acknowledged	Acknowledged	Not acknowledged
Multiplexing	No	Yes	Yes
Bandwidth utilization	Poor	Moderate	High

Another class of service is *class F*, which is used for fabric management. Class F is similar to Class 2 and provides notification of nondelivery of frames.

Fabric Services

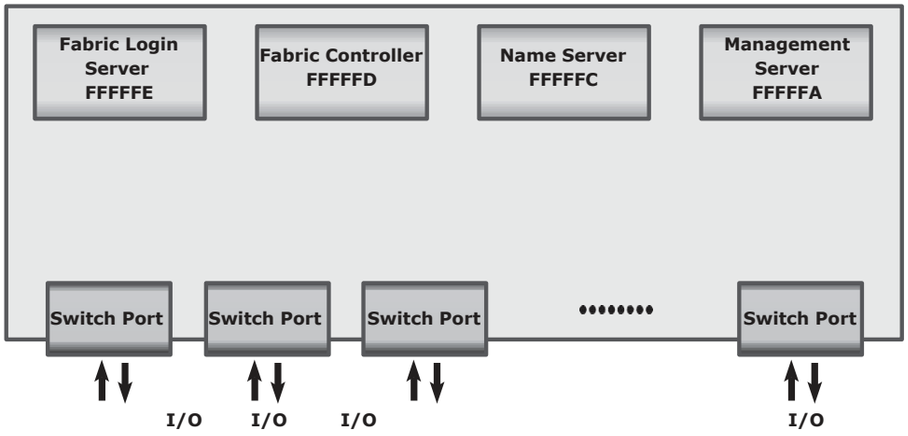
All FC switches, regardless of the manufacturer, provide a common set of services as defined in the Fibre Channel standards. These services are available at certain predefined addresses. Some of these services are Fabric Login Server, Fabric Controller, Name Server, and Management Server (see - 5-16).

The *Fabric Login Server* is located at the predefined address of FFFFFE and is used during the initial part of the node's fabric login process.

The *Name Server* (formally known as *Distributed Name Server*) is located at the predefined address FFFFFC and is responsible for name registration and management of node ports. Each switch exchanges its Name Server information with other switches in the fabric to maintain a synchronized, distributed name service.

Each switch has a *Fabric Controller* located at the predefined address FFFFFD. The Fabric Controller provides services to both node ports and other switches. The Fabric Controller is responsible for managing and distributing Registered State Change Notifications (RSCNs) to the node ports registered with the

Fabric Controller. If there is a change in the fabric, RSCNs are sent out by a switch to the attached node ports. The Fabric Controller also generates Switch Registered State Change Notifications (SW-RSCNs) to every other domain (switch) in the fabric. These RSCNs keep the name server up-to-date on all switches in the fabric.



- 5-16: Fabric services provided by FC switches

FFFFFAistheFibreChanneladdressforthe*ManagementServer*.TheManagem ent Server is distributed to every switch within the fabric. The Management Server enables the FC SAN management software to retrieve information and admin- ister the fabric.

Switched Fabric Login Types

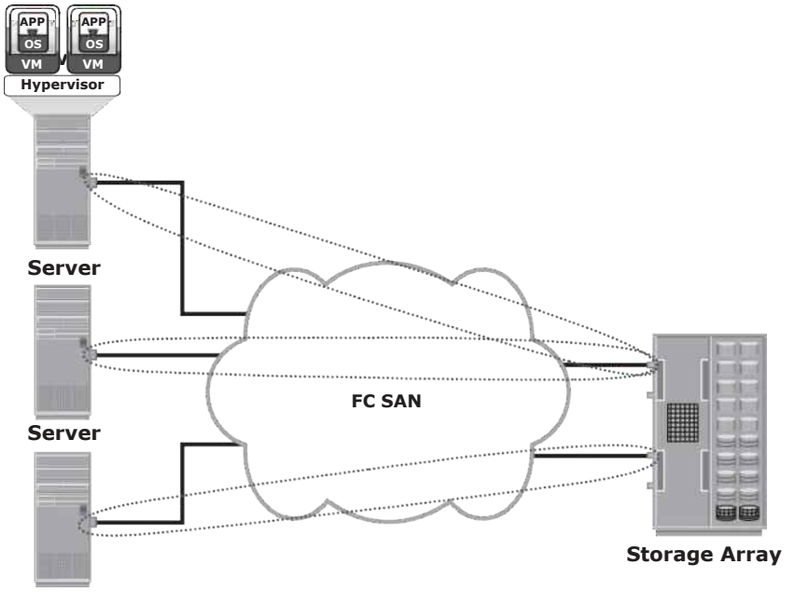
Fabric services define three login types:

- ***Fabric login* (FLOGI):** Performed between an N_Port and an F_Port. To log on to the fabric, a node sends a FLOGI frame with the WWNN and WWPN parameters to the login service at the predefined FC address FFFFFE (Fabric Login Server). In turn, the switch accepts the login and returns an Accept (ACC) frame with the assigned FC address for the node. Immediately after the FLOGI, the N_Port registers itself with the local Name Server on the switch, indicating its WWNN, WWPN, port type, class of service, assigned FC address and so on. After the N_Port has logged in, it can query the name server database for information about all other logged in ports.

- ***Port login* (PLOGI):** Performed between two N_Ports to establish asession. The initiator N_Port sends a PLOGI request frame to the target N_Port, which accepts it. Thetarget N_Port returns an ACC to the initiator N_Port. Next, the N_Portsexchangeservice parametersrelevanttothesession.
- ***Process login* (PRLI):** Also performed between two N_Ports. This login relatesto the FC-4 ULPs, such as SCSI. If the ULP is SCSI, N_Ports exchange SCSI-related service parameters.

Zoning

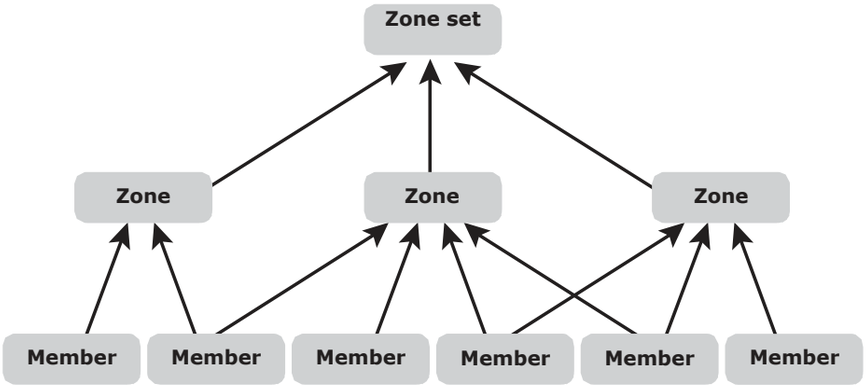
Zoning is an FC switch function that enables node ports within the fabric to be logically segmented into groups and to communicate with each other within the group (see - 5-17).



- 5-17: Zoning

Whenever a change takes place in the name server database, the fabric controller sends a Registered State Change Notification (RSCN) to all the nodes impacted by the change. If zoning is not configured, the fabric controller sends an RSCN to all the nodes in the fabric. Involving the nodes that are not impacted by the change results in increased fabric-management traffic. For

a large fabric, the amount of FC traffic generated due to this process can be significant and might impact the host-to-storage data traffic. Zoning helps to limit the number of RSCNs in a fabric. In the presence of zoning, a fabric sends the RSCN to only those nodes in a zone where the change has occurred. Zone members, zones, and zone sets form the hierarchy defined in the zoning process (see - 5-18). A *zone set* is composed of a group of zones that can be activated or deactivated as a single entity in a fabric. Multiple zone sets may be defined in a fabric, but only one zone set can be active at a time. *Members* are nodes within the SAN that can be included in a zone. Switch ports, HBA ports, and storage device ports can be members of a zone. A port or node can be a member of multiple zones. Nodes distributed across multiple switches in a switched fabric may also be grouped into the same zone. Zone sets are also referred to as *zone configurations*.



- 5-18: Members, zones, and zone sets

Zoning provides control by allowing only the members in the same zone to establish communication with each other.

Types of Zoning

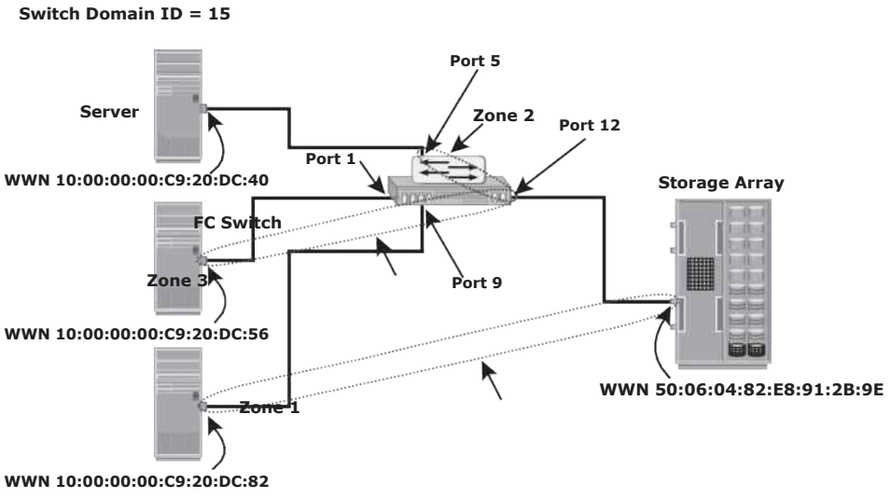
Zoning can be categorized into three types:

- **Port zoning:** Uses the physical address of switch ports to define zones. In port zoning, access to node is determined by the physical switch port to which a node is connected. The zone members are the port identifier (switchdomain ID and port number) to which HBA and itstargets (storage devices) are connected. If a node is moved to another switch port in the

fabric, then zoning must be modified to allow the node, in its new port, to participate in its original zone. However, if an HBA or storage device port fails, an administrator just has to replace the failed device without changing the zoning configuration.

- **WWN zoning:** Uses World Wide Names to define zones. The zone members are the unique WWN addresses of the HBA and its targets (storage devices). Amajoradvantage of WWNzoningis itsflexibility. WWN zoning allows nodes to be moved to another switch port in the fabric andmain- tain connectivity to its zone partners without having to modify the zone configuration. This is possible because the WWN is static to the node port.
- **Mixed zoning:** Combines the qualities of both WWN zoning and port zoning. Using mixed zoning enables a specific node port to be tied to the WWN of another node.

- 5-19 shows the three types of zoning on an FC network.



- 5-19: Types of zoning

Zoning is used with LUNmasking to control serveraccessto storage. However, these are two different activities. Zoning takes place at the fabric level and LUN masking is performed at the array level.

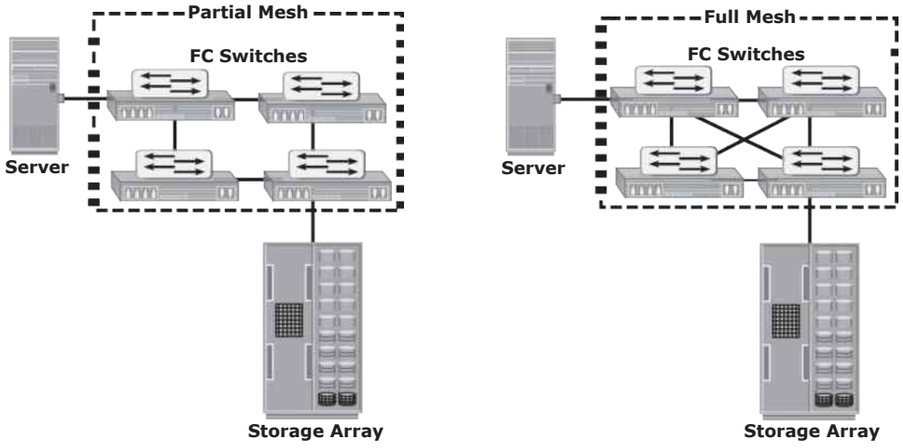
FC SAN Topologies

Fabric design follows standard topologies to connect devices. Core-edge fabric is one of the popular topologies for fabric designs. Variations of core-edge fabric and mesh topologies are most commonly deployedin FC SANimplementations.

Mesh Topology

A mesh topology may be one of the two types: full mesh or partial mesh. In a *full mesh*, every switch is connected to every other switch in the topology. A full mesh topology may be appropriate when the number of switches involved is small. A typical deployment would involve up to four switches or directors, with each of them servicing highly localized host-to-storage traffic. In a full mesh topology, a maximum of one ISL or hop is required for host-to-storage traffic. However, with the increase in the number of switches, the number of switch ports used for ISL also increases. This reduces the available switch ports for node connectivity.

In a *partial mesh topology*, several hops or ISLs may be required for the traf- fic to reach its destination. Partial mesh offers more scalability than full mesh topology. However, without proper placement of host and storage devices, traf- fic management in a partial mesh fabric might be complicated and ISLs could become overloaded due to excessive traffic aggregation. -5-20 depicts both partial mesh and full mesh topologies.



- 5-20: Partial mesh and full mesh topologies

Core-Edge Fabric

The *core-edge fabric* topology has two types of switch tiers. The *edge tier* is usually composed of switches and offers an inexpensive approach to adding more hosts in a fabric. Each switch at the edge tier is attached to a switch at the core tier through ISLs.

The *core tier* is usually composed of enterprise directors that ensure high fabric availability. In addition, typically all traffic must either traverse this tier or termi- nate at this tier. In this configuration, all storage devices are connected to the core tier, enabling host-to-storage traffic to traverse only one ISL. Hosts that require high performance may be connected directly to the core tier and consequently avoid ISL delays.

In *core-edge topology*, the edge-tier switches are not connected to each other. The core-edge fabric topology increases connectivity within the SAN while conserving the overall port utilization. If fabric expansion is required, additional

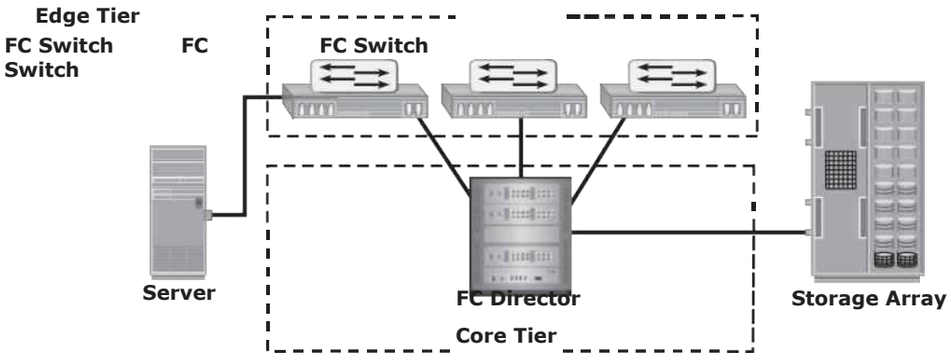
edge switches are connected to the core. The core of the fabric is also extended by adding more switches or directors at the core tier. Based on the number of

core-tier switches, this topology has different variations, such as, *single-core topology* (see - 5-21) and *dual-core topology* (see - 5-22). To transform a single-core topology to dual-core, new ISLs are created to connect each edge switch to the new core switch in the fabric.

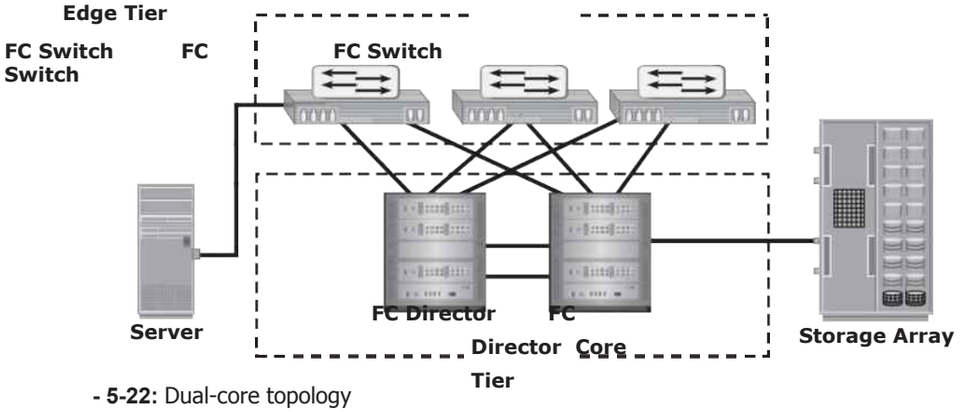
Benefits and Limitations of Core-Edge Fabric

The core-edge fabric provides maximum one-hop storage access to all storage devices in the system. Because traffic travels in a deterministic pattern (from the edge to the core and vice versa), a core-edge provides easier calculation of the ISL load and traffic patterns. In this topology, because each tier's switch port

is used for either storage or hosts, it's easy to identify which network resources are approaching their capacity, making it easier to develop a set of rules for scaling and apportioning.



- 5-21: Single-coretopology

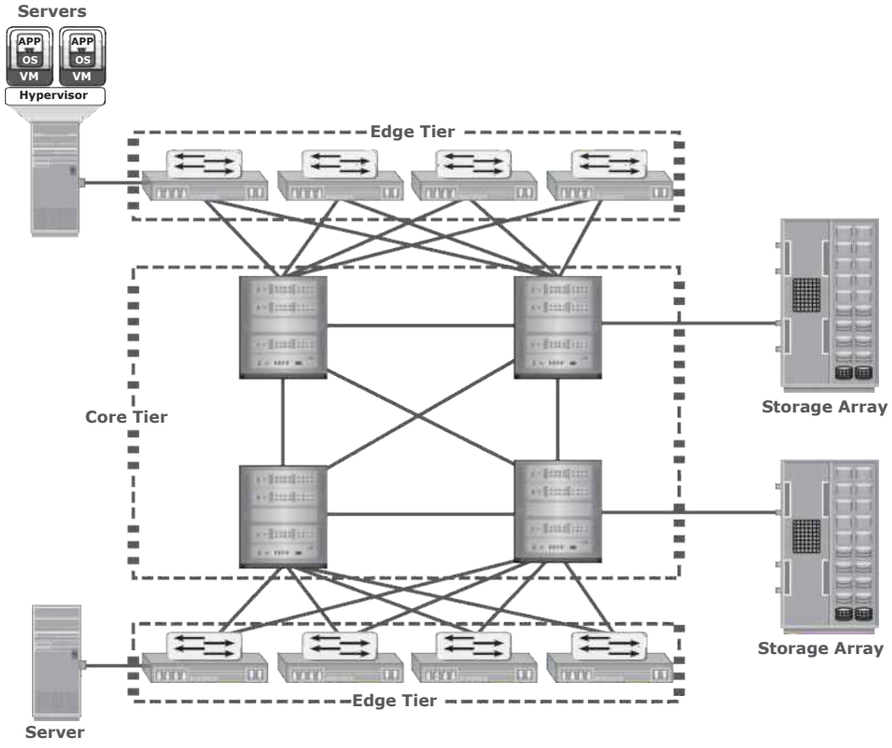


- 5-22: Dual-core topology

Core-edge fabrics are scaled to larger environments by adding more core switches and linking them, or adding more edge switches. This method enables extending the existing simple core-edge model or expanding the fabric into a compound or complex core-edge model.

However, the core-edge fabric might lead to some performance-related prob- lems because scaling a core-edge topology involves increasing the number of hop counts in the fabric. *Hop count* represents the total number of ISLs traversed by a packet between its source and destination. A common best practice is to keep the number of host-to-storage hops unchanged, at one hop, in a core-edge. Generally, a large hop count means a high data transmission delay between the source and destination.

As the number of cores increases, it is prohibitive to continue to maintain ISLs from each core to each edge switch. When this happens, the fabric design is changed to a compound or complex core-edge design (see - 5-23).



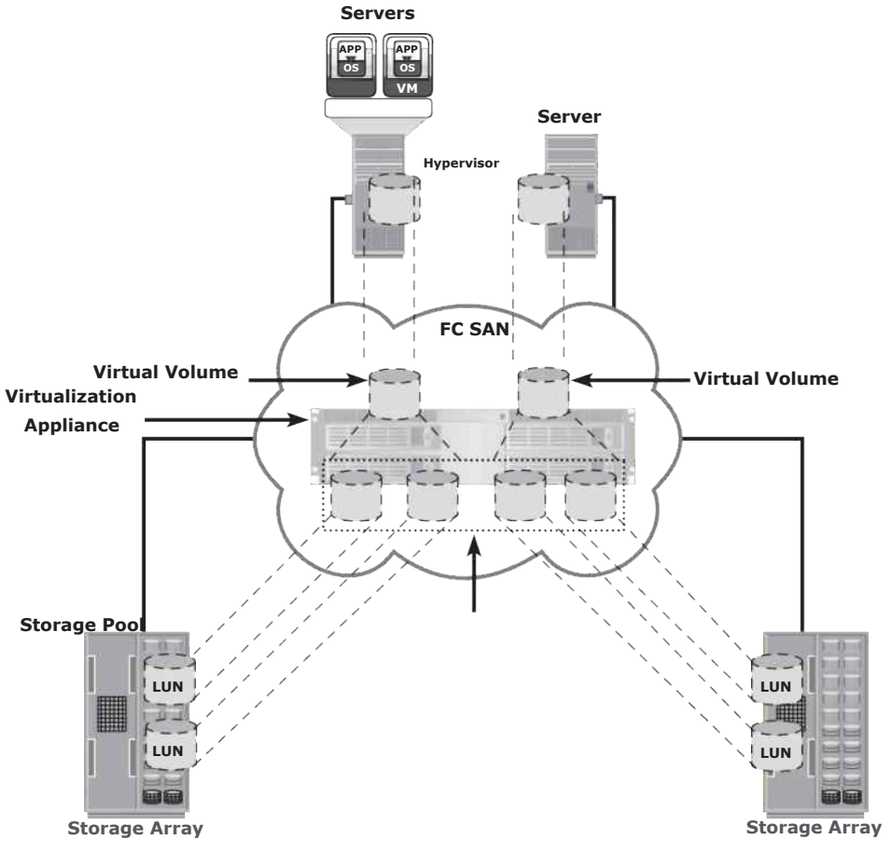
- 5-23: Compound core-edge topology

Virtualization in SAN

This section details two network-based virtualization techniques in a SAN environment: block-level storage virtualization and virtual SAN (VSAN).

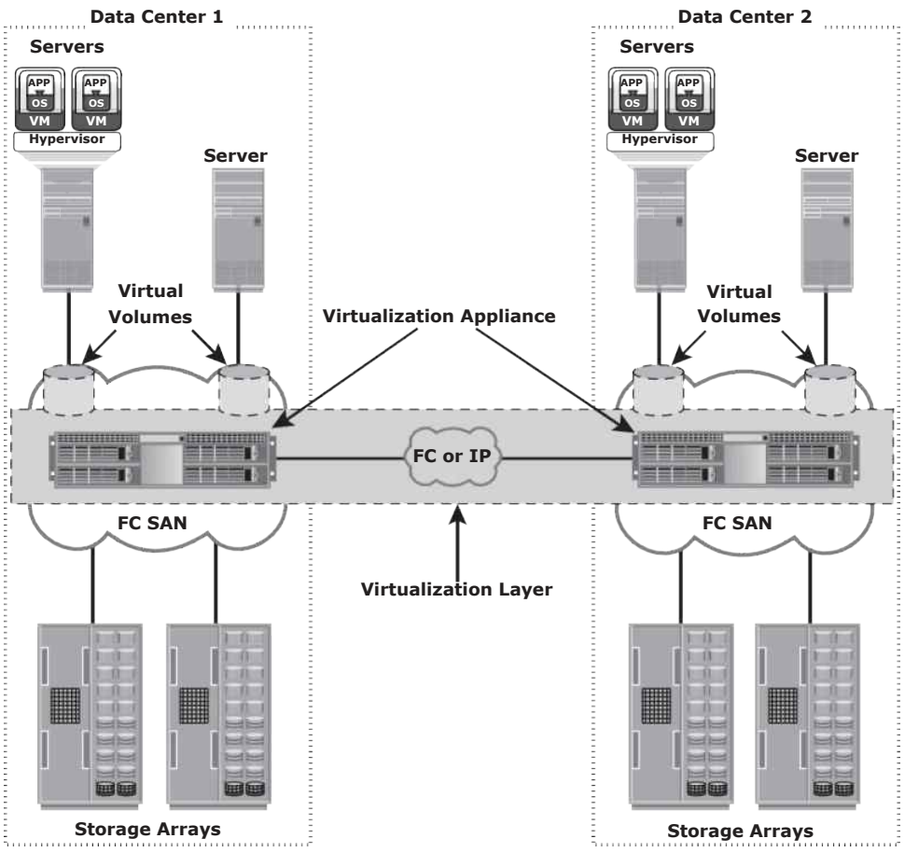
Block-level Storage Virtualization

Block-level storage virtualization aggregates block storage devices (LUNs) and enables provisioning of virtual storage volumes, independent of the underlying physical storage. A virtualization layer, which exists at the SAN, abstracts the identity of physical storage devices and creates a storage pool from heterogeneous storage devices. Virtual volumes are created from the storage pool and assigned to the hosts. Instead of being directed to the LUNs on the individual storage arrays, the hosts are directed to the virtual volumes provided by the virtualization layer. For hosts and storage arrays, the virtualization layer appears as the target and initiator devices, respectively. The virtualization layer maps the virtual volumes to the LUNs on the individual arrays. The hosts remain unaware of the mapping operation and access the virtual volumes as if they were accessing the physical storage attached to them. Typically, the virtualization layer is managed via a dedicated virtualization appliance to which the hosts and the storage arrays are connected.



- 5-24: Block-level storage virtualization

Previously, block-level storage virtualization provided non disruptive data migration only within a data center. The new generation of block-level storage virtualization enables nondisruptive data migration both within and between data centers. It provides the capability to connect the virtualization layers at multiple data centers. The connected virtualization layers are managed centrally and work as a single virtualization layer stretched across data centers (see - 5-25). This enables the federation of block-storage resources both within and across data centers. The virtual volumes are created from the federated storage resources.



- 5-25: Federation of block storage across data centers

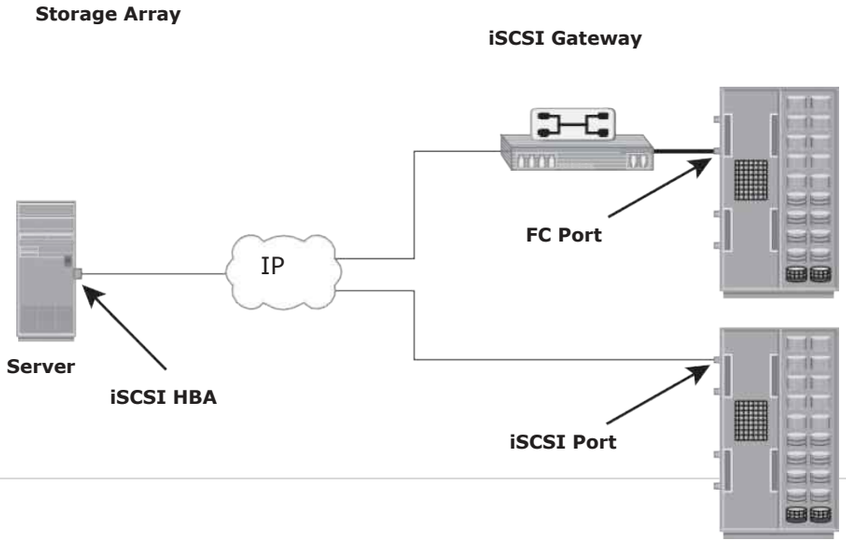
Virtual SAN (VSAN)

Virtual SAN (also called *virtual fabric*) is a logical fabric on an FC SAN, which enables communication among a group of nodes regardless of their physical location in the fabric. In a VSAN, a group of hosts or storage ports communicate with each other using a virtual topology defined on the physical SAN. Multiple VSANsmay be created on a single physical SAN. Each VSAN acts as an indepen- dent fabric with its own set of fabric services, such as name server, and zoning. Fabric-related configurations in one VSAN do not affect the traffic in another. VSANs improve SAN security, scalability, availability, and manageability.

VSANsprovide enhanced security by isolating the sensitive data in a VSAN and by restricting access to the resources located within that VSAN. The same Fibre Channel addresscan be assigned to nodesindifferent VSANs, thus increasing the fabric scalability. Events causing traffic disruptions in one VSAN are contained

iSCSI

iSCSI is an IP based protocol that establishes and manages connections between host and storage overIP,as shownin - 6-1. iSCSI encapsulates SCSI commands and data into an IP packet and transports them using **TCP/IP. iSCSI is widely adopted for connecting servers to storage because it is relatively inexpensive and easy to implement**, especially in environments in which an FC SAN does not exist.



- 6-1: iSCSI implementation

Components of iSCSI

An initiator (host), target (storage or iSCSI gateway), and an IP-based network are the key iSCSI components.

If an iSCSI-capable storage array is deployed, then a host with the iSCSI initiator can directly communicate with the storage array over an IP network. However, in an implementation that uses an existing FC array for iSCSI communication, an iSCSI gatewayis used. These devices perform

the translation of IP packets to FC frames and vice versa, thereby bridging the connectivity between the IP and FC environments.

iSCSI Host Connectivity

A standard NIC with software iSCSI initiator, a TCP offload engine (TOE) NIC with software iSCSI initiator, and an iSCSI HBA are the three iSCSI host connectivity options. The function of the iSCSI initiator is to route the SCSI commands over an IP network.

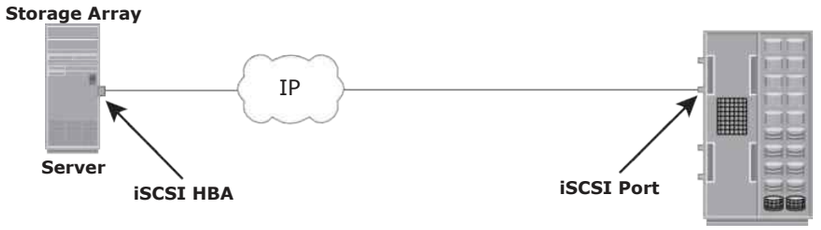
iSCSI Topologies

Two topologies of iSCSI implementations are native and bridged. *Native topology* does not have FC components. The initiators may be either directly attached to targets or connected through the IP network. *Bridged topology* enables the coexistence of FC with IP by providing iSCSI-to-FC bridging functionality. For example, the initiators can exist in an IP environment while the storage remains in an FC environment.

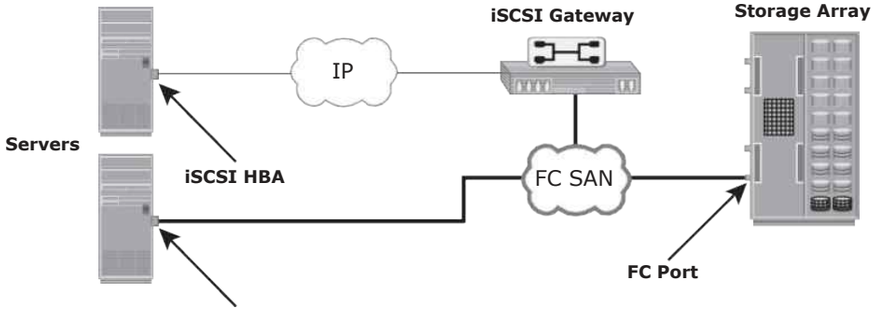
Native iSCSI Connectivity

FC components are not required for iSCSI connectivity if an iSCSI-enabled array is deployed. In - 6-2 (a), the array has one or more iSCSI ports configured with an IP address and is connected to a standard Ethernet switch.

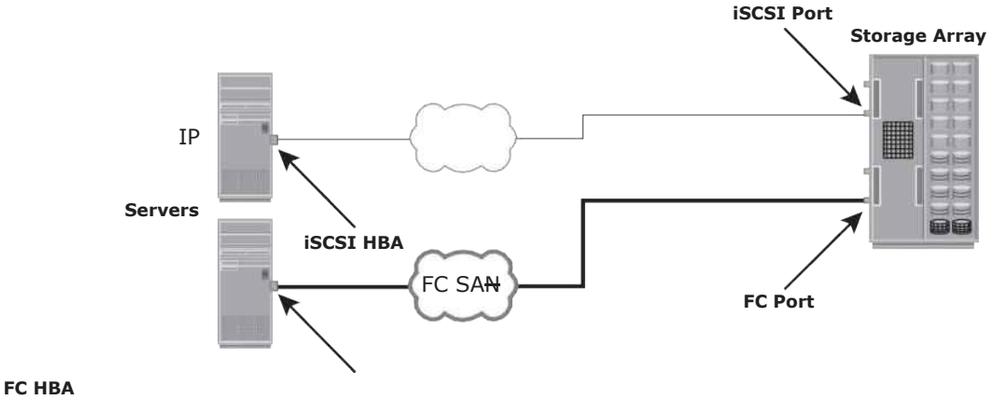
After an initiator is logged on to the network, it can access the available LUNs on the storage array. A single array port can service multiple hosts or initiators as long as the array port can handle the amount of storage traffic that the hosts generate.



(a) Native iSCSI Connectivity



(b) Bridged iSCSI Connectivity



(c) Combining FC and Native iSCSI Connectivity

- 6-2: iSCSITopologies

Bridged iSCSI Connectivity

A bridged iSCSI implementation includes FC components in its configuration. - 6-2 (b) illustrates iSCSI host connectivity to an FC storage array.

In this case, the array does not have any iSCSI ports. Therefore, an external device, called a gateway or a multiprotocol router, must be used to facilitate the communication between the iSCSI host and FC storage. The gateway converts IP packets to FC frames and vice versa. The bridge devices contain both FC and Ethernet ports to facilitate the communication between the FC and IP environments.

In a bridged iSCSI implementation, the iSCSI initiator is configured with the gateway's IP address as its target destination. On the other side, the gateway is configured as an FC initiator to the storage array.

Combining FC and Native iSCSI Connectivity

The most common topology is a combination of FC and native iSCSI. Typically, a storage array comes with both FC and iSCSI ports that enable iSCSI and FC connectivity in the same environment, as shown in - 6-2 (c).

iSCSI Protocol Stack

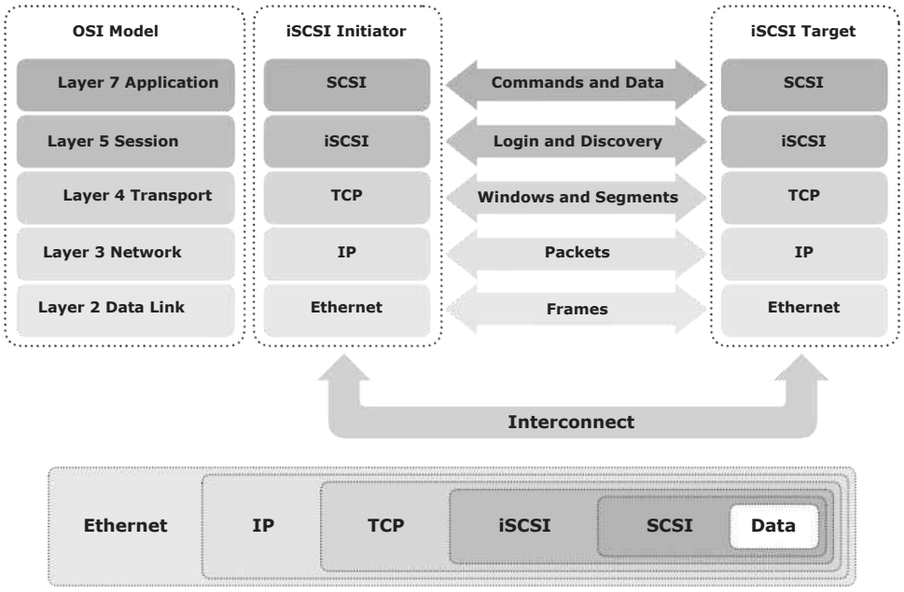
- 6-3 displays a model of the iSCSI protocol layers and depicts the encapsulation order of the SCSI commands for their delivery through a physical carrier.

SCSI is the command protocol that works at the application layer of the Open System Interconnection (OSI) model. The initiators and targets use SCSI commands and responses to talk to each other. The SCSI command descriptor blocks, data, and status messages are encapsulated into TCP/IP and transmitted across the network between the initiators and targets.

iSCSI is the session-layer protocol that initiates a reliable session between devices that recognize SCSI commands and TCP/IP.

The iSCSI session-layer interface is responsible for handling login, authentication, target discovery, and session management. TCP is used with iSCSI at the transport layer to provide reliable transmission.

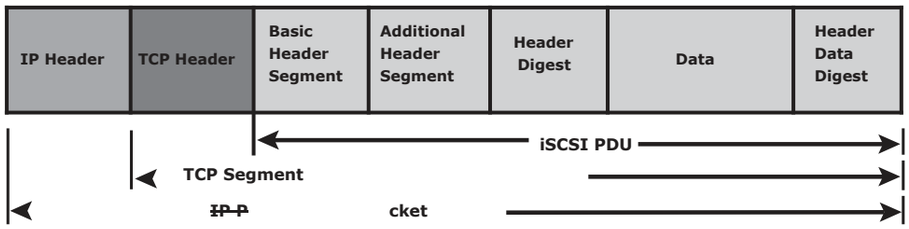
TCP controls message flow, windowing, error recovery, and retransmission. It relies upon the network layer of the OSI model to provide global addressing and connectivity. The Layer 2 protocols at the data link layer of this model enable node-to-node communication through a physical network.



- 6-3: iSCSI protocol stack

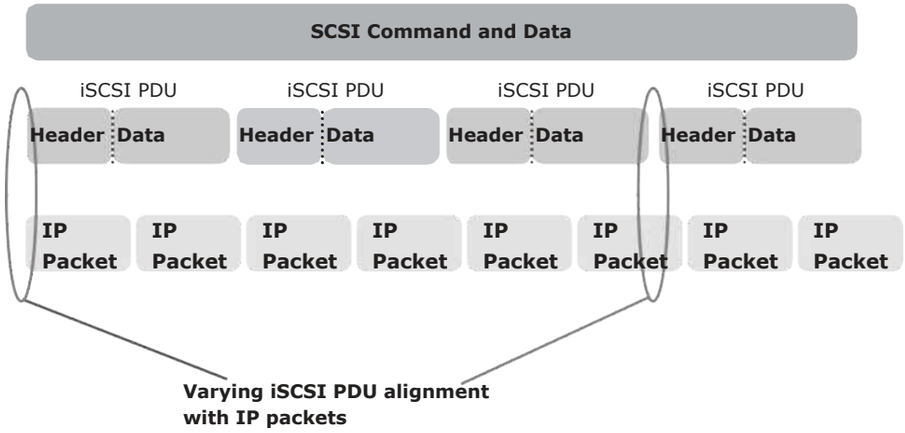
iSCSI PDU

A *protocol data unit* (PDU) is the basic —information unitll in the iSCSI environment. The iSCSI initiators and targets communicate with each other using iSCSI PDUs. This communication includes establishing iSCSI connections and iSCSI sessions, performing iSCSI discovery, sending SCSI commands and data, and receiving SCSI status. All iSCSI PDUs contain one or more header segments followed by zero or more data segments. The PDU is then encapsulated into an IP packet to facilitate the transport.



- 6-4: iSCSI PDU encapsulated in an IP packet

A message transmitted on a network is divided into a number of packets. If necessary, each packet can be sent by a different route across the network. Packets can arrive in a different order than the order in which they were sent. IP only delivers them; it is up to TCP to organize them in the right sequence. The target extracts the SCSI commands and data on the basis of the information in the iSCSI header.



- 6-5: Alignment of iSCSI PDUs with IP packets

To achieve the 1:1 relationship between the IP packet and the iSCSI PDU, the maximum transmission unit (MTU) size of the IP packet is modified. This eliminates fragmentation of the IP packet, which improves the trans- mission efficiency.

iSCSI Discovery

An initiator must discover the location of its targets on the network and the names of the targets available to it before it can establish a session. This discov- ery can take place in two ways: *SendTargets discovery* or *internet Storage Name Service* (iSNS).

In *SendTargets discovery*, the initiator is manually configured with the target's network portal to establish a discovery session. The initiator issues the `SendTargets` command, and the target network portal responds with the names and addresses of the targets available to the host.

iSNS (see - 6-6) enables automatic discovery of iSCSI devices on an IP network. The initiators and targets can be configured to automatically register themselves with the iSNS server. Whenever an initiator wants to know the targets that it can access, it can query the iSNS server for a list of available targets.

The discovery can also take place by using service location protocol (SLP). However, this is less commonly used than `SendTargets` discovery and iSNS.

iSCSI Names

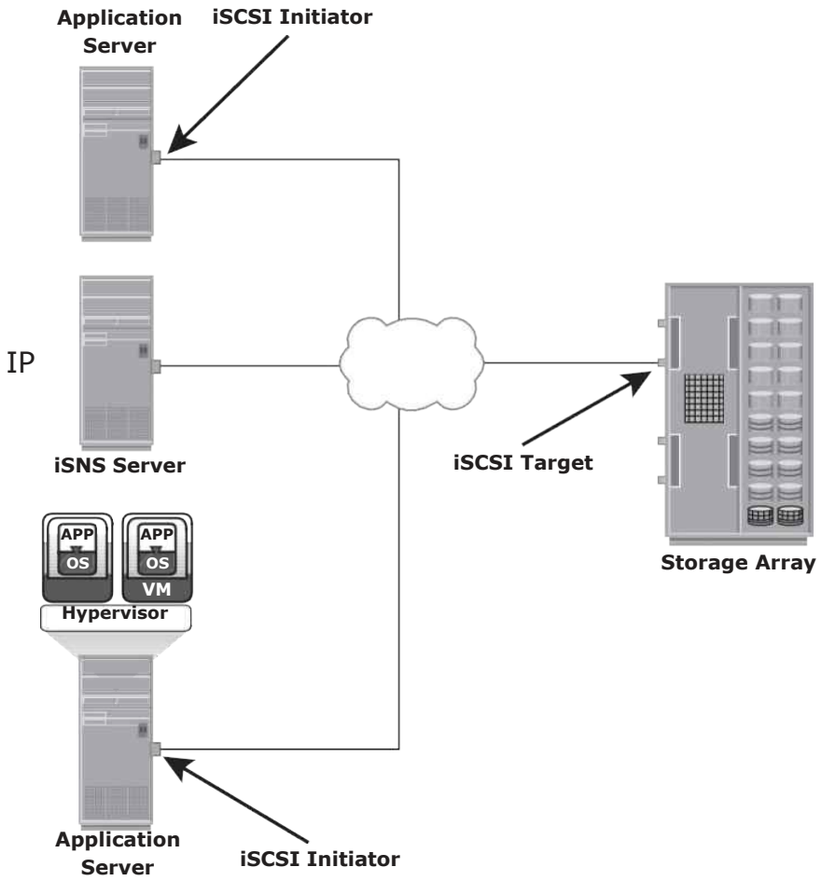
A unique worldwide iSCSI identifier, known as an *iSCSI name*, is used to identify the initiators and targets within an iSCSI network to facilitate communication.

iSCSI Qualified Name (IQN): An organization must own a registered domain name to generate iSCSI Qualified Names. This domain name does not need to be active or resolve to an address. It just needs to be reserved to prevent other organizations from using the same domain name to generate iSCSI names. A date is included in the name to avoid potential conflicts caused by the transfer of domain names. An example of an IQN is `iqn.2008-02.com.example:`*`optional_string`*.

The *optional_string* provides aserial number, an asset number, or any other device identifiers. An iSCSI Qualified Name enables storageadministra- tors to assign meaningful names to iSCSI devices, and therefore, manage those devices more easily.

- **Extended Unique Identifier (EUI):** An EUI is a globally unique identifier based on the IEEE EUI-64 naming standard. An EUI is composed of the eui prefix followed by a 16-character hexadecimal name, such as `eui.0300732A32598D26`.

In either format, the allowed special characters are dots, dashes, and blank spaces.



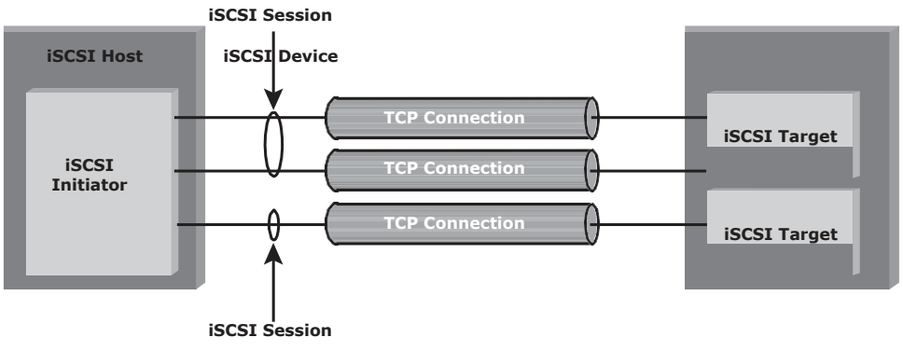
- 6-6: Discovery using iSNS

iSCSI Session

An iSCSI session is established between an initiator and a target, as shown. A sessionis identified by a session ID (SSID), which includes part of an initiator ID and a target ID. The session can be intended for one of the following:

- Thediscoveryof the available targetsbytheinitiatorsandthe locationof a specific target on a network
- The normal operation of iSCSI (transferring data between initiators and targets)

There might be one or more TCP connections within each session. Each TCP connection within the session has a unique connection ID (CID).



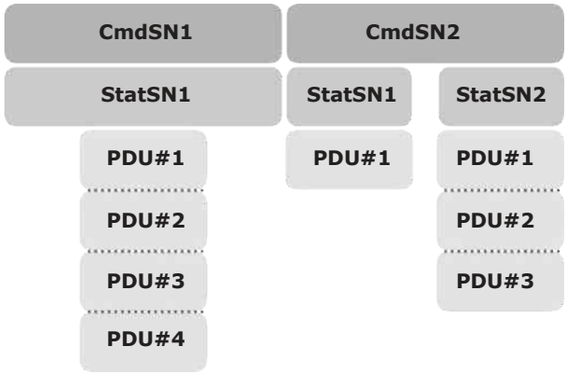
An iSCSI session is established via the iSCSI login process..

iSCSI Command Sequencing

The iSCSI communication between the initiators and targets is based on the request-response command sequences. A command sequence may generate multiple PDUs. A *command sequence number* (CmdSN) within an iSCSI session is used for numbering all initiator-to-target command PDUs belonging to the session. This number ensures that every command is delivered in the same order in which it is transmitted, regardless of the TCP connection that carries the command in the session.

Command sequencing begins with the first login command, and the CmdSN is incremented by one for each subsequent command.

Similar to command numbering, a *status sequence number* (StatSN) is used to sequentially number status responses,



Command and status sequence number

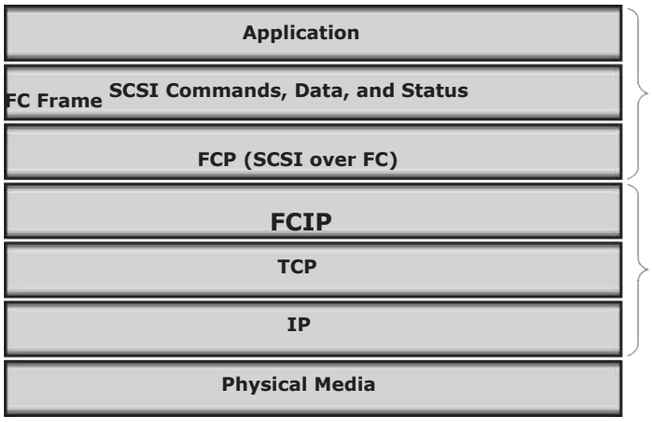
FCIP

FC SAN provides a high-performance infrastructure for localized data movement. Organizations are now looking for ways to transport data over a long distance between their disparate SANs at multiple geographic locations. FCIP is a tunneling protocol that enables distributed FC SAN is lands to be interconnected over the existing IP-based networks.

FCIP might require high network bandwidth when replicating or backing up data. FCIP does not handle data traffic throttling or flow control; these are controlled by the communicating FC switches and devices within the fabric.

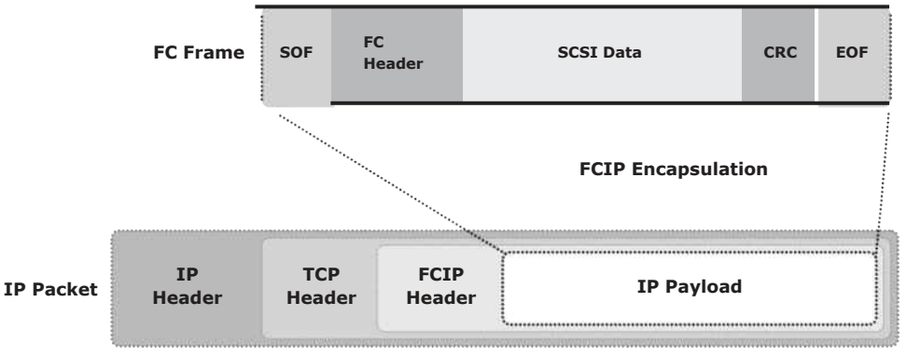
FCIP Protocol Stack

The FCIP protocol stack is shown in -: 6-9. Applications generate SCSI commands and data, which are processed by various layers of the protocol stack.



: FCIP protocol stack

The upper layer protocol SCSI includes the SCSI driver program that executes the read-and-writecommands. Belowthe SCSI layeristhe Fibre Channel Protocol (FCP) layer, which is simply a Fibre Channel frame whose payload is SCSI. The FCP layer rides on top of the Fibre Channel transport layer. This enables the FC frames to run natively within a SAN fabric environment. In addition, the FC frames can be encapsulated into the IP packet and sent to a remote SAN over the IP. The FCIP layer encapsulates the Fibre Channel frames onto the IP payload and passes them to the TCP layer (see - 6-10). TCP and IP are used for transporting the encapsulated information across Ethernet, wireless, or other media that support the TCP/IP traffic.



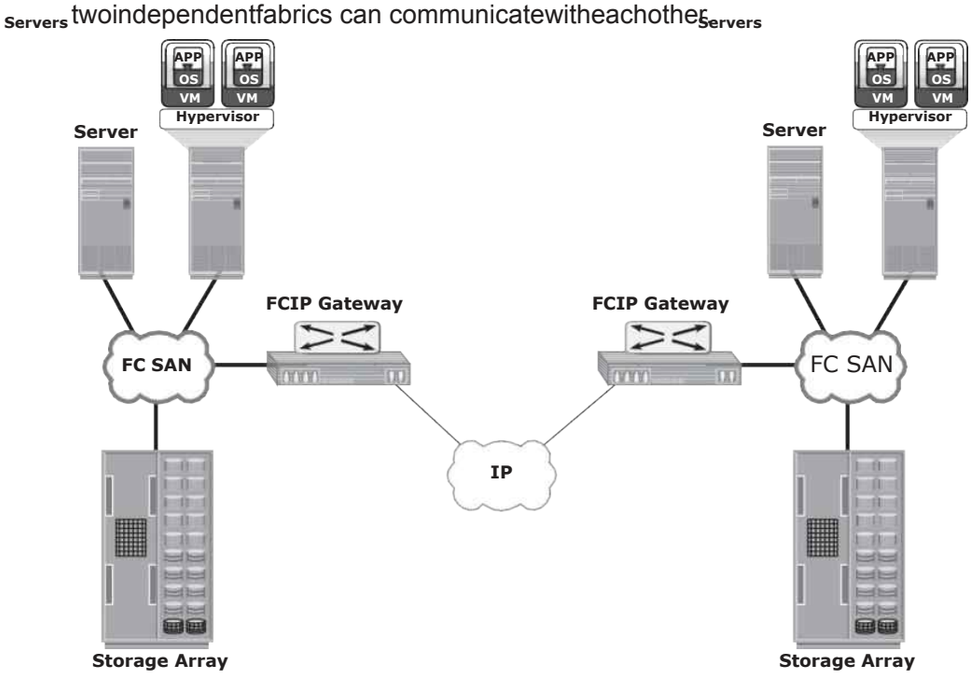
- 6-10: FCIP encapsulation

Encapsulation of FC frame into an IP packet could cause the IP packet to be fragmented whenthe data link cannot support the maximum transmission unit

(MTU) size of an IP packet. When an IP packet is fragmented, the required parts of the header must be copied by all fragments. When a TCP packet is segmented, normal TCP operations are responsible for receiving and re-sequencing the data prior to passing it on to the FC processing portion of the device.

FCIP Topology

In an FCIPenvironment, an FCIPgatewayisconnected to eachfabricviaa standard FC connection (see - 6-11). The FCIP gateway at one end of the IP network encapsulatesthe FC frames into IP packets. The gateway at the other end removes the IP wrapper and sends the FC data to the layer 2 fabric. The fabric treats these gateways as layer 2 fabric switches. An IP address is assigned to the port on the gateway, which is connected to an IP network. After the IP connectivity is estab- lished, the nodesinthe twoindependentfabrics can communicatewitheachother.



- 6-11: FCIP topology

FCIP Performance and Security

Performance, reliability, and security should always be taken into consideration whenimplementingstoragesolutions. The implementation of FCIP is also subject to the same considerations.

Fromthe perspective of performance, configuring multiplepathsbetween FCIP gateways eliminates single points of failure and provides increased bandwidth. In a scenario of extended distance, the IP network might be a bottleneck if sufficient bandwidth is not available. In addition, because FCIP creates a unified fabric, disruption in the underlying IP network can cause instabilities in the SAN environment. These instabilities include a segmented fabric, excessive RSCNs, and host timeouts.

The vendors of FC switches have recognized some of the drawbacks related to FCIP and have implemented features to enhance stability, such as the capability to segregate the FCIP traffic into a separate virtual fabric.

Security is also a consideration in an FCIP solution because the data is trans- mitted over public IP channels. Various security options are available to protect

the data based on the router's support. IPSec is one such security measure that

can be implemented in the FCIP environment.

FCoE

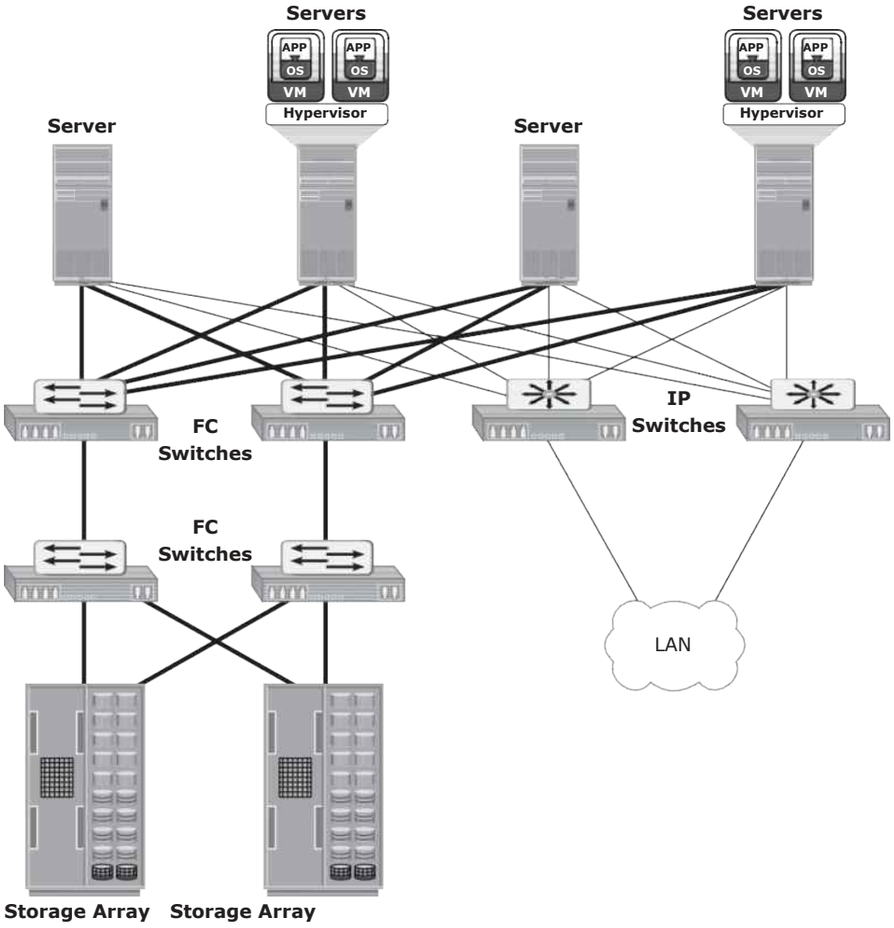
Data centers typically have multiple networks to handle various types of I/O traffic — for example, an Ethernet network for TCP/IP communication and an FC network for FC communication. TCP/IP is typically used for client-server communication, data backup, infrastructure management communication, and so on. FC is typically used for moving block-level data between storage and servers. To support multiple networks, servers in a data center are equipped with multiple redundant physical network interfaces — for example, multiple Ethernet and FC cards/adapters. In addition, to enable the communication, different types of networking switches and physical cabling infrastructure are implemented in data centers. The need for two different kinds of physi- cal network infrastructure increases the overall cost and complexity of data center operation.

Fibre Channel over Ethernet (FCoE) protocol provides consolidation of LAN and SAN traffic over a single physical interface infrastructure. FCoE helps organizations address the challenges of having multiple discrete network infra- structures. FCoE uses the Converged Enhanced Ethernet (CEE) link (10 Gigabit Ethernet) to send FC frames over Ethernet.

I/O Consolidation Using FCoE

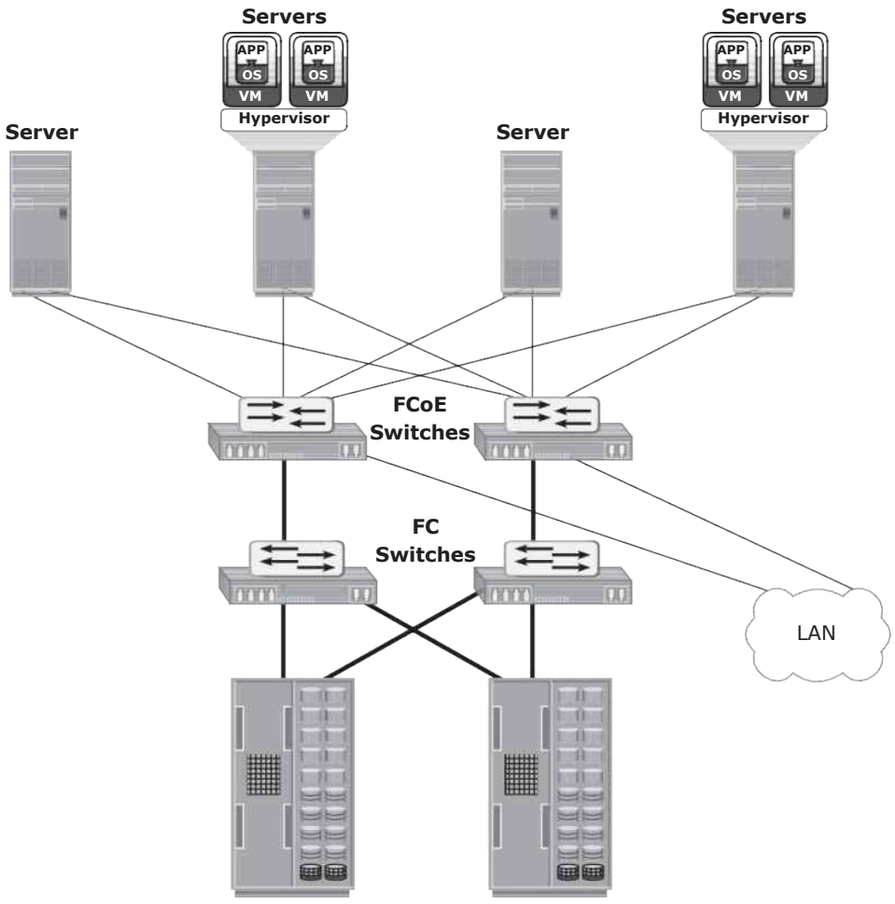
The key benefit of FCoE is I/O consolidation. - 6-12 represents the infra- structure before FCoE deployment. Here, the storage resources are accessed using HBAs, and the IP network resources are accessed using NICs by the servers. Typically, in a data center, a server is configured with 2 to 4 NIC cards and redundant HBA cards. If the data center has hundreds of servers, it would

require a large number of adapters, cables, and switches. This leads to a complex environment, which is difficult to manage and scale. The cost of power, cooling, and floor space further adds to the challenge.



- 6-12: Infrastructure before using FCoE

- 6-13 shows the I/O consolidation with FCoE using FCoE switches and Converged Network Adapters (CNAs). A CNA (discussed in the section —Converged Network Adapter‖) replaces both HBAs and NICs in the server and consolidates both the IP and FC traffic. This reduces the requirement of multiple
network adapters at the server to connect to different networks. Overall, this reduces the requirement of adapters, cables, and switches. This also consider-
ably reduces the cost and management overhead.



- 6-13: Infrastructure after using FCoE

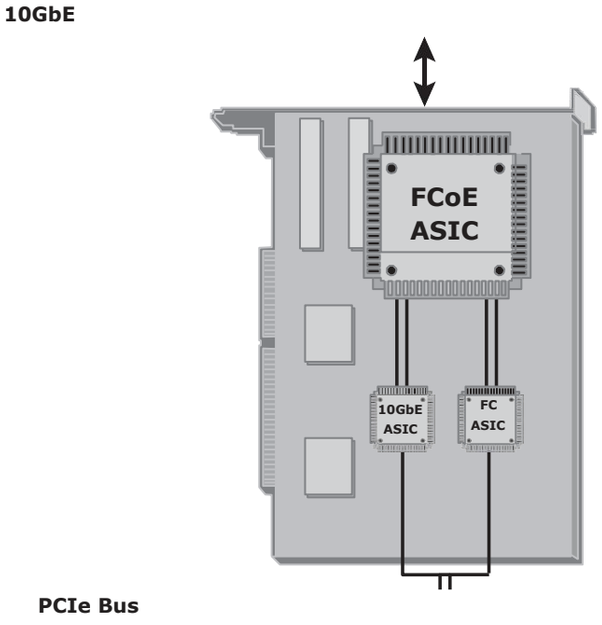
Components of an FCoE Network

This section describes the key physical components required to implement FCoE in a data center. The key FCoE components are:

- Converged Network Adapter (CNA)
- Cables
- FCoE switches

Converged Network Adapter

A CNA provides the functionality of both a standard NIC and an FC HBA in a single adapter and consolidates both types of traffic. CNA eliminates the need to deploy separate adapters and cables for FC and Ethernet communications, thereby reducing the required number of server slots and switch ports. CNA offloads the FCoE protocol processing task from the server, thereby freeing the server CPU resources for applicationprocessing. As shown in - 6-14, a CNA contains separate modules for 10 Gigabit Ethernet, Fibre Channel, and FCoE Application Specific Integrated Circuits (ASICs). The FCoE ASIC encapsulates FC frames into Ethernet frames. One end of this ASIC is connected to 10GbE and FC ASICs for server connectivity, while the other end provides a 10GbE interface to connect to an FCoE switch.



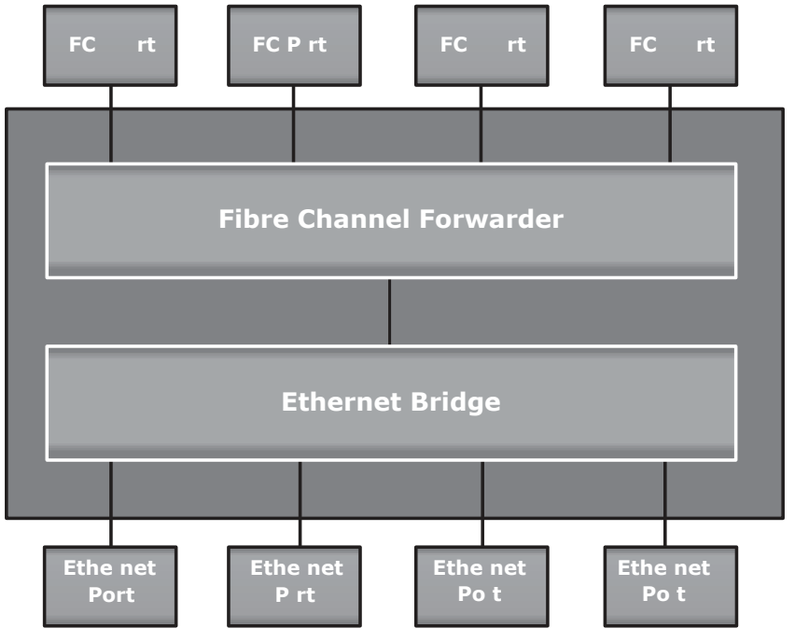
- 6-14: Converged Network Adapter

Cables

Currently two options are available for FCoE cabling: Copper based Twinax and standard fiber optical cables. A Twinax cable is composed of two pairs of copper cables covered with a shielded casing. The Twinax cable can transmit data at the speed of 10 Gbps over shorter distances up to 10 meters. Twinax cables require less power and are less expensive than fiber optic cables. The Small Form Factor Pluggable Plus (SFP+) connector is the primary connector used for FCoE links and can be used with both optical and copper cables.

FCoE Switches

An FCoE switch has both Ethernet switch and Fibre Channel switch functionalities. The FCoE switch has a Fibre Channel Forwarder (FCF), Ethernet Bridge, and set of Ethernet ports and optional FC ports, as shown in - 6-15. The function of the FCF is to encapsulate the FC frames, received from the FC port, into the FCoE frames and also to de-encapsulate the FCoE frames, received from the Ethernet Bridge, to the FC frames.

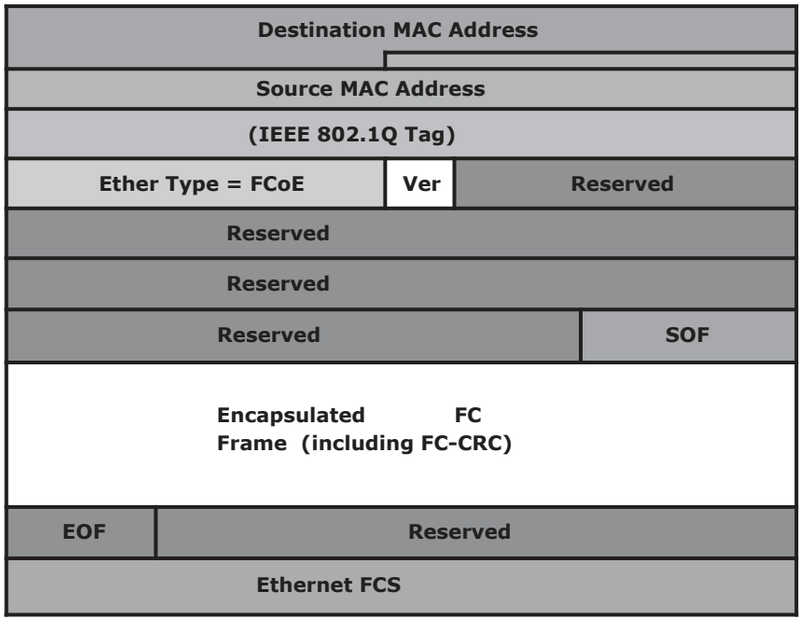


- 6-15: FCoE switch generic architecture

Upon receiving the incoming traffic, the FCoE switch inspects the Ethertype (used to indicate which protocol is encapsulated in the payload of an Ethernet frame) of the incoming frames and uses that to determine the destination. If the Ethertype of the frame is FCoE, the switch recognizes that the frame contains an FC payload and forwards it to the FCF. From there, the FC is extracted from the FCoE frame and transmitted to FC SAN over the FC ports. If the Ethertype is not FCoE, the switch handles the traffic as usual Ethernet traffic and forwards it over the Ethernet ports.

FCoE Frame Structure

An FCoE frame is an Ethernet frame that contains an FCoE Protocol Data Unit. - 6-16 shows the FCoE frame structure. The first 48-bits in the frame are used to specify the destination MAC address, and the next 48-bits specify the source MAC address. The 32-bit IEEE 802.1Qtag supportsthe creation of multiple virtual networks (VLANs) across a single physical infrastructure. FCoE has its own Ethertype, as designated by the next 16 bits, followed by the 4-bit version field. The next 100-bits are reserved and are followed by the 8-bit Start of Frame and then the actual FC frame. The 8-bit End of Frame delimiter is followed by 24 reserved bits. The frame ends with the final 32-bits dedicated to the Frame Check Sequence (FCS) function that provides error detection for the Ethernet frame.



- 6-16: FCoE frame structure

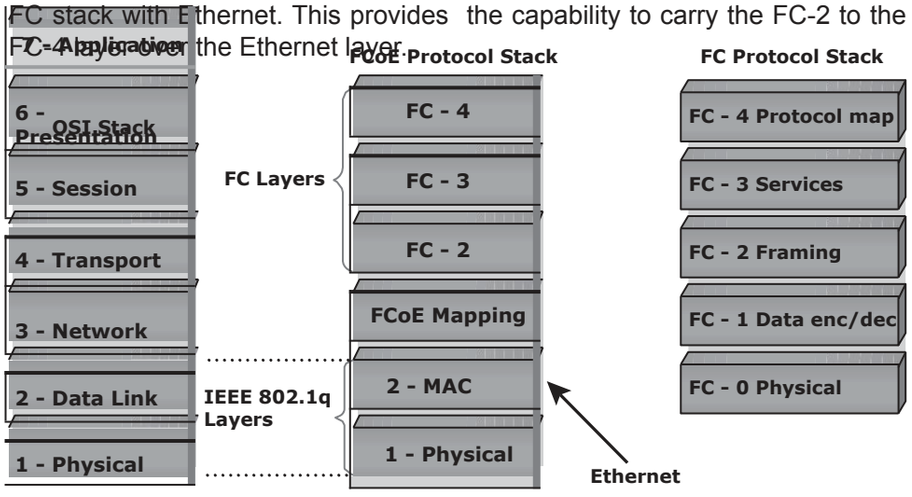
The encapsulated Fibre Channel frame consists of the original 24-byte FC header and the data being transported (including the Fibre Channel CRC). The FC frame structure is maintained such that when a traditional FC SAN is connected to an FCoE capable switch, the FC frame is de-encapsulated from the FCoE frame and transported to FC SAN seamlessly. This capability enables FCoE to integrate with the existing FC SANs without the need for a gateway.

Frame size is also an important factor in FCoE. A typical Fibre Channel data frame has a 2,112-byte payload, a 24-byte header, and an FCS. A standard Ethernet frame has a default payload capacity of 1,500 bytes. To maintain good

performance, FCoE must use jumbo frames to prevent a Fibre Channel frame from being split into two Ethernet frames. The next chapter discusses jumbo frames in detail. FCoE requires Converged Enhanced Ethernet, which provides lossless Ethernet and jumbo frame support.

FCoE Frame Mapping

The encapsulation of the Fibre Channel frame occurs through the mapping of the FC frames onto Ethernet, as shown in - 6-17. Fibre Channel and traditional networks have stacks of layers where each layer in the stack represents a set of functionalities. The FC stack consists of five layers: FC-0 through FC-4. Ethernet is typically considered as a set of protocols that operates at the physical and data link layers in the seven layer OSI stack. The FCoE protocol specification replaces the FC-0 and FC-1 layers of the FC stack with Ethernet. This provides the capability to carry the FC-2 to the FC-4 layer over the Ethernet layer.



- 6-17: FCoE frame mapping

FCoE Enabling Technologies

Conventional Ethernet is lossy in nature, which means that frames might be dropped or lost during transmission. *Converged Enhanced Ethernet* (CEE), or lossless Ethernet, provides a new specification to the existing Ethernet standard that eliminates the lossy nature of Ethernet. This makes 10 Gb Ethernet a viable storage networking option, similar to FC. Lossless Ethernet requires certain functionalities. These functionalities are defined and maintained by the data center bridging (DCB) task group, which is a part of the IEEE 802.1 working group, and they are:

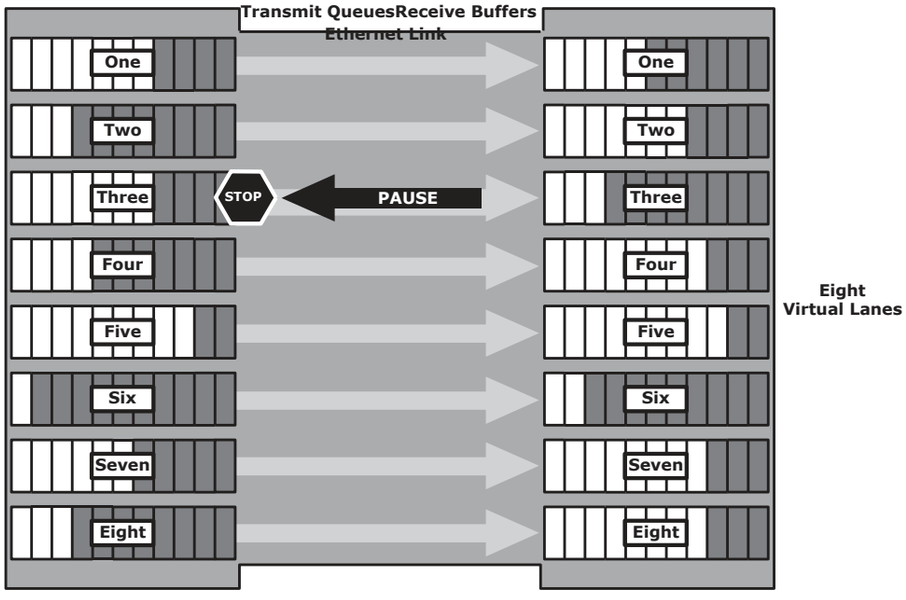
- Priority-based flow control
- Enhanced transmission selection

- Congestion Notification
- Data center bridging exchange protocol

Priority-Based Flow Control (PFC)

Traditional FC manages congestion through the use of a link-level, credit-based flow control that guarantees no loss of frames. Typical Ethernet, coupled with TCP/IP,uses apacketdrop flow controlmechanism. The packetdrop flow control is not lossless. This challenge is eliminated by using an IEEE 802.3x Ethernet PAUSE control frame to create a lossless Ethernet. A receiver can send a PAUSE request to a sender when the receiver's buffer is filling up. Upon receiving a PAUSE frame, the sender stops transmitting frames, which guarantees no loss of frames. The downside of using the Ethernet PAUSE frame is that it operates on the entire link, which might be carrying multiple traffic flows.

PFC provides a link level flow control mechanism. PFC creates eight separate virtual links on a single physical link and allows any of these links to be paused and restarted independently. PFC enables the pause mechanism based on user priorities or classes of service. Enabling the pause based on priority allows cre- ating lossless links for traffic, such as FCoE traffic. This PAUSE mechanism is typically implemented for FCoE while regular TCP/IP traffic continues to drop frames. - 6-18 illustrates how a physical Ethernet link is divided into eight virtual links and allows a PAUSE for a single virtual link without affecting the traffic for the others.



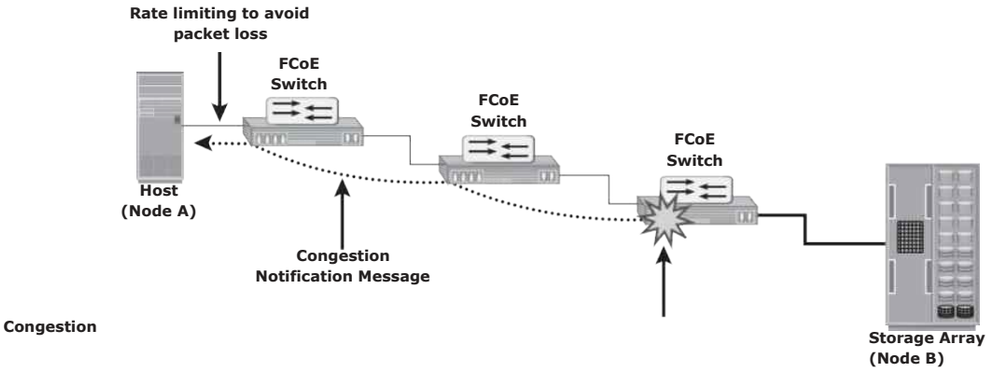
- 6-18: Priority-based flow control

Enhanced Transmission Selection (ETS)

Enhanced transmission selection provides a common management framework for the assignment of bandwidth to different traffic classes, such as LAN, SAN, and Inter Process Communication (IPC). When a particular class of traffic does not use its allocated bandwidth, ETS enables other traffic classes to use the available bandwidth.

Congestion Notification (CN)

Congestion notification provides end-to-end congestion management for protocols, such as FCoE, that do not have built-in congestion control mechanisms. Link level congestion notification provides a mechanism for detecting congestion and notifying the source to move the traffic flow away from the congested links. Link level congestion notification enables a switch to send a signal to other ports that need to stop or slow down their transmissions. The process of congestion notification and its management is shown in - 6-19, which represents the communication between the nodes A (sender) and B (receiver). If congestion at the receiving end occurs, the algorithm running on the switch generates a congestion notification message to the sending node (Node A). In response to the CN message, the sending end limits the rate of data transfer.



- 6-19: Congestion Notification

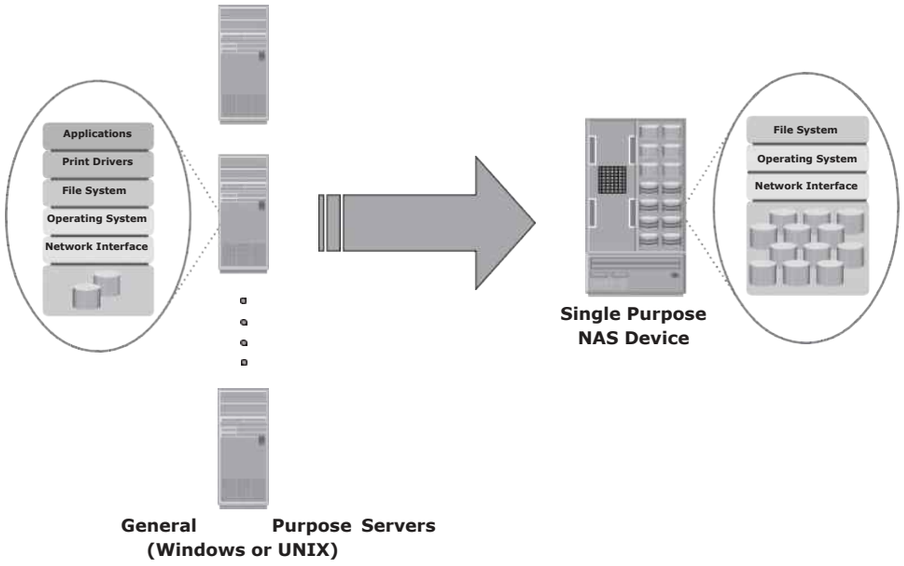
Data Center Bridging Exchange Protocol (DCBX)

DCBX protocol is a discovery and capability exchange protocol, which helps Converged Enhanced Ethernet devices to convey and configure their features with the other CEE devicesin the network. DCBX is used to negotiate capabilities

between the switches and the adapters, and it allows the switch to distribute the configuration values to all the attached adapters. This helps to ensure consistent configuration across the entire network.

General-Purpose Servers versus 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 - 7-1, a general-purpose server can be used to host any application because it runs a general-purpose operating system. Unlike a general-purpose server, a NAS device is dedicated to file-serving. It has specialized operat- ing system dedicated to file serving by using industry-standard protocols. Some NAS vendors support features, such as native clustering for high availability.



- 7-1: General purpose server versus NAS device

Benefits of NAS

NAS offers the following benefits:

- **Comprehensiveaccesstoinformation:** Enablesefficientfilesharingand supports many-to-one and one-to-many configurations. Themany-to-one configuration enables a NAS device to serve many clients simultaneously. Theone-to-manyconfigurationenablesoneclienttoconnectwithmany NAS devices simultaneously.
- **Improved efficiency:** NAS delivers better performance compared to a general-purpose fileserverbecause NAS usesanoperatingsystem specialized for file serving.
- **Improved flexibility:** Compatible with clients on both UNIX and Windows platforms usingindustry-standard protocols. NAS isflexibleandcan serve requests from different types of clients from the same source.
- **Centralized storage:** Centralizes data storage tominimize 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

- **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 tech- nology for failover.
- **Security:** Ensuressecurity, userauthentication, 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.

File Systems and Network File Sharing

A *file system* is a structured way to store and organize data files. Many file systems maintain a file access table to simplify the process of searching and accessing files.

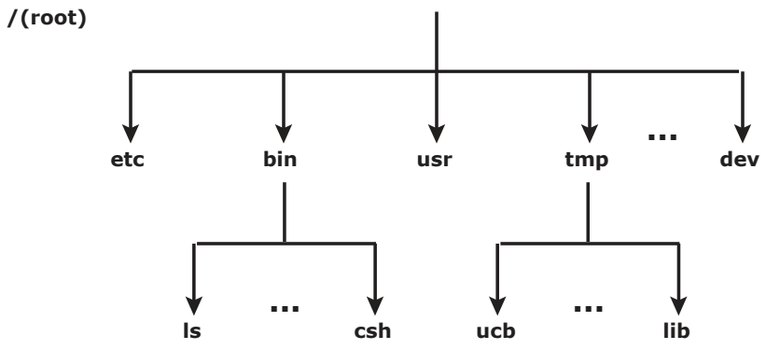
Accessing a File System

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

Network File Sharing

Network file sharing refers to storing and accessing files over a network. In a file-sharing environment, the user who creates a file (the creator or owner of a file) determines the type of access (such as read, write, execute, append, and

delete) to be given to other users and controls changes to the file. When multiple users try to access a shared file at the same time, a locking scheme is required to maintain data integrity and, at the same time, make this sharing possible.



- 7-2: UNIX directory structure

Some examples of file-sharing methods are file transfer protocol (FTP), Distributed File System(DFS), client-servermodelsthat use file-sharing protocols such as NFS and CIFS, and the peer-to-peer (P2P) model

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. When FTP is used over SSH, it is referred to as Secure FTP (SFTP).

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. Standard client-server file- sharing protocols, such as 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. Using this protocol, the clients mount remote file systems that are available on dedicated file servers.

A name service, such as Domain Name System (DNS), and directory services such as Microsoft Active Directory, and Network Information Services (NIS), helps users identify and access a unique resource over the network. A *name*

service protocol such as the Lightweight Directory Access Protocol (LDAP) creates a namespace, which holds the unique name of every network resource and helps

recognize resources on the network.

A *peer-to-peer* (P2P) file sharing model uses a peer-to-peer network. P2P enables client machines to directly share files with each other over a

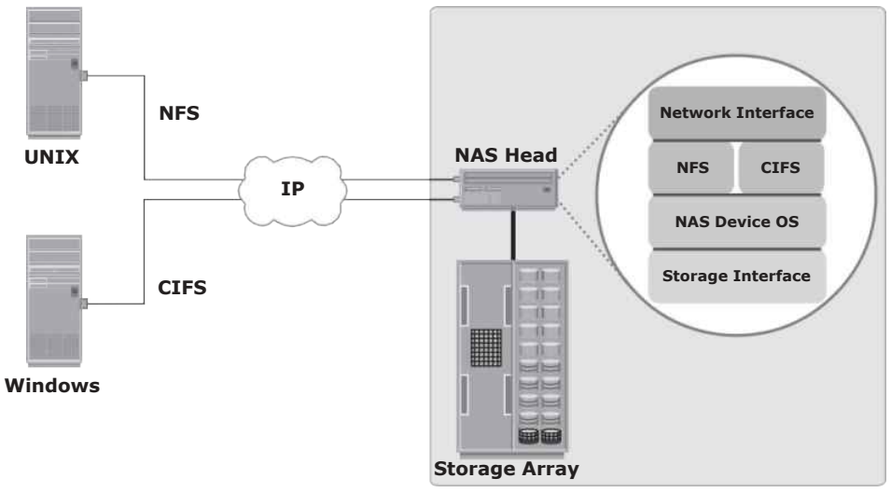
network. Clients use a file sharing software that searches for other peer clients. This differs from the client-server model that uses file servers to store files for sharing.

Components of NAS

A NAS device has two key components: NAS head and storage (see - 7-3). In some NAS implementations, the storage could be external to the NAS device and shared with other hosts. The NAS head includes the following components:

- CPU and memory
- Oneormorenetworkinterface cards(NICs), whichprovideconnectivity totheclient network. Examplesofnetwork protocolssupportedby 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.
- NFS, CIFS, and other protocols for file sharing
- Industry-standard storage protocols and ports to connect andmanage physical disk resources

The NAS environment includes clients accessing a NAS device over an IP network using file-sharing protocols.



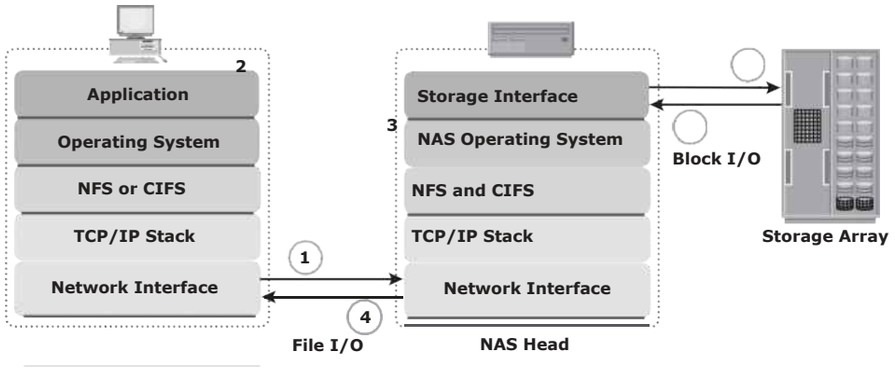
- 7-3: Components of NAS

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. 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 storagerequest,whichisablock-levelI/O,andthenperformstheoperation 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.

- 7-4 illustrates this process.



- 7-4: NAS I/O operation

NAS Implementations

Three common NAS implementations are unified, gateway, and scale-out. The *unified* NAS consolidates NAS-based and SAN-baseddataaccesswithin a unified storage platform and provides a unified management interface for managing both the environments.

In a *gateway* implementation, the 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.

The *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

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. Due to consolidation of NAS-based and SAN-based access on a single storage platform, unified NAS reduces an organization's infrastructure and management costs.

A unified NAS contains one or more NAS heads and storagein 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.

Unified NAS Connectivity

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.

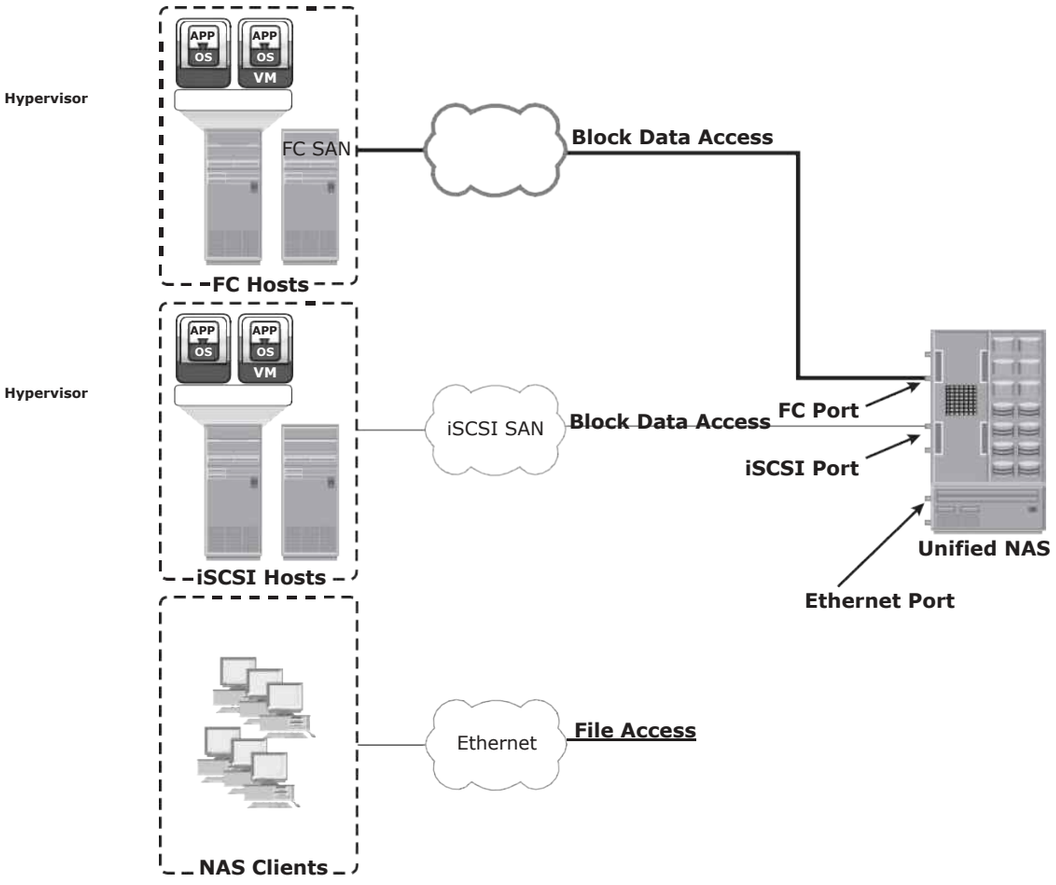
iSCSI and FC ports on a storage controller enable hosts to access the storage directly or through a storage network at the block level. - 7-5 illustrates an example of unified NAS connectivity.

Gateway NAS

A gateway NAS device consists of one or more NAS heads and uses external and independently managed storage. Similar to unified NAS, the storage is shared with other applications that use block-level I/O. 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.

For example, NAS heads can be added to scale up the NAS device performance. When the storage limit is reached, it can scale up, adding capacity on the SAN, independent of NAS heads. Similar to a unified NAS, a gateway NAS also enables high utilization of storage capacity by sharing it with the SAN environment.

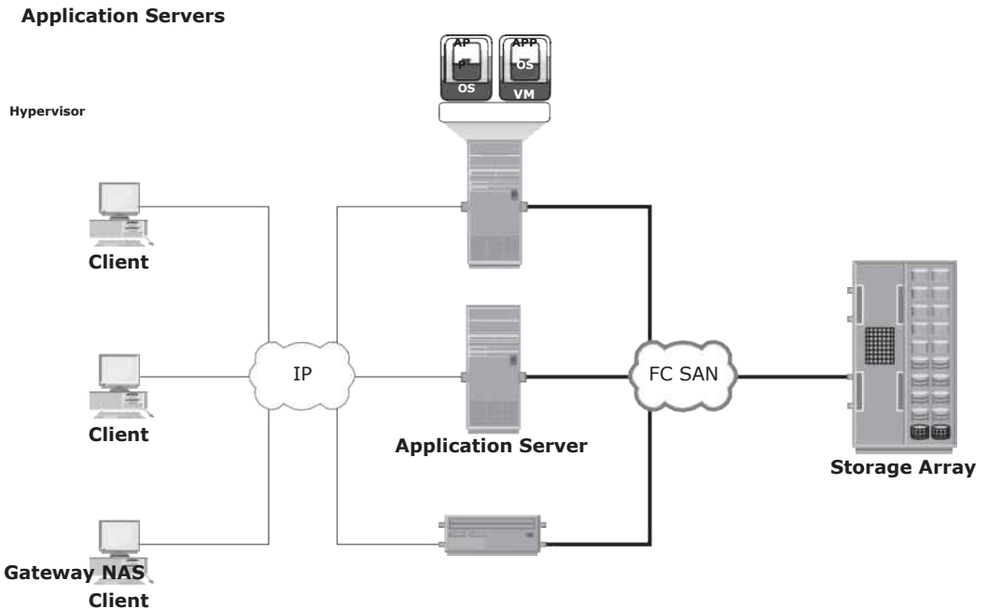


- 7-5: Unified NAS connectivity

Gateway NAS Connectivity

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 systemin a gatewaysolutionis achievedthrough atraditional FC SAN. To deploy a gateway NAS solution, factors, such as multiple paths for data, redundant

fabrics, and load distribution, must be considered. - 7-6 illustrates an example of gateway NAS connectivity.



- 7-6: Gateway NAS connectivity

Implementation of both unified and gateway solutions requires analysis of the SAN environment. This analysis is required to determine the feasibil- ity of combining the NAS workload with the SAN workload. Analyze the SAN to determine whether the workload is primarily read or write, and if it is random or sequential. Also determine the predominant I/O size in use. Typically, NAS workloads are random with small I/O sizes. Introducing sequential workload with random workloads can be disruptive to the sequen- tial 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 system.

Scale-Out NAS

Both unified and gateway NAS implementations provide the capability to scale- up their resources based on data growth and rise in performance requirements. Scaling up these NAS devices involves adding CPUs, memory, and storage to

the NAS device. Scalability is limited by the capacity of the NAS device to house and use additional NAS heads and storage.

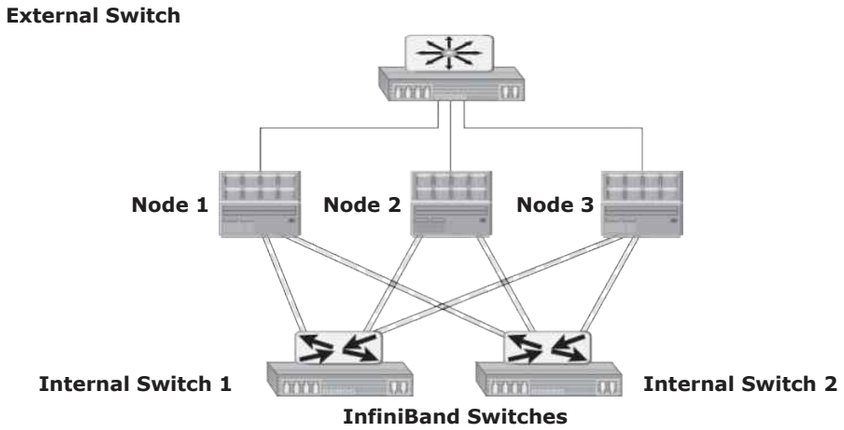
Scale-out NAS enables grouping multiple nodes together to construct a clustered NAS system. A scale-out NAS provides the capability to scale its resources by simply adding nodes to a clustered NAS architecture. The cluster works as a single NAS device and is managed centrally. Nodes can be added to the cluster, when more performance or more capacity is needed, without causing any downtime. Scale-out NAS provides the flex- ibility to use many nodes of moderate performance and availability char- acteristics to produce a total system that has better aggregate performance and availability. It also provides ease of use, low cost, and theoretically unlim- ited scalability.

Scale-out NAS creates a single file system that runs on all nodes in the cluster. All information is shared among nodes, so the entire file system is accessible by clients connecting to any node in the cluster. Scale-out NAS stripes data across all nodes in a cluster along with mirror or parity pro- tection. As data is sent from clients to the cluster, the data is divided and allocated to different nodes in parallel. When a client sends a request to read a file, the scale-out NAS retrieves the appropriate blocks from mul- tiple nodes, recombines the blocks into a file, and presents the file to the client. As nodes are added, the file system grows dynamically and data is evenly distributed to every node. Each node added to the cluster increases the aggregate storage, memory, CPU, and network capacity. Hence, cluster performance also increases.

Scale-out NAS is suitable to solve the —Big Datall challenges that enterprises and customers face today. It provides the capability to manage and store large, high-growth data in a single place with the flexibility to meet a broad range of performance requirements.

Scale-Out NAS Connectivity

Scale-out NAS clusters use separate internal and external networks for back-end and front-end connectivity, respectively. An internal network provides con- nections for intracluster communication, and an external network connection enables clients to access and share file data. Each node in the cluster connects to the internal network. 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. - 7-7 illustrates an example of scale-out NAS connectivity.



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

NAS File-Sharing Protocols

Most NAS devices support multiple file-service protocols to handle file I/O requests to a remote file system. As discussed earlier, 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.

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. 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 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.

Currently, three versions of NFS are in use:

- **NFSversion2(NFSv2):** UsesUDPtoprovideastatelessnetworkconnectionbetweenaclientandaserver. Features, suchaslocking, arehandled 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 file size, asynchronous writes, and additional file attributes to reduce refetching.
- **NFSversion4(NFSv4):**UsesTCPandisbasedonastatefulprotocoldesign. 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.

(Continued)

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. CIFS provides the following

features to ensure data integrity:

- ltusesfileandrecordlockingtopreventusersfromoverwritingthework of another user on a file or arecord.
- 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 applicationis written totake advan- tage of these features. Moreover, CIFS is a stateful protocol because the CIFS server maintains connection information regarding every connected

client. If anetworkfailure or CIFS server failure occurs, the client receivesa 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 must 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`.

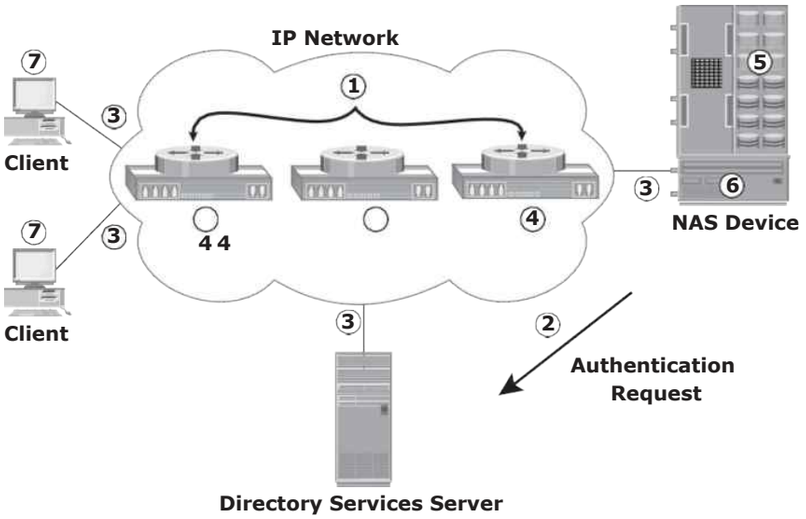
The file naming scheme in an NFS environment is:
`Server:/export` or **`Server.domain.suffix:/export`.**

Factors Affecting NAS Performance

NAS uses IP network; therefore, bandwidth and latency issues associated with IP affect NAS performance. Network congestion is one of the most significant sources of latency (- 7-8) in a NAS environment. Other factors that affect NAS performance at different levels follow:

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 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.
3. **Retransmission:** Link errors and buffer overflows can result in retransmis- sion. 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.
4. **Overutilized routers and switches:** The amount of time that an over-utilized 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 ina network. Additionaldevicesshould be added if the current devices are overutilized.

5. **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 direc- tory structures and can be resolved by flattening the directory structure. Poor file system layout and an overutilized disk system can also degrade performance.
6. **Over utilized NAS devices:** Clients accessing multiple files 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 becaused bya poor filesystem structure or insufficient resources in a storage subsystem.
7. **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.



- 7-8: Causes of latency

Configuring *virtual LANs* (VLANs), setting proper Maximum Transmission Unit (MTU) and TCP window sizes, and link aggregation can improve NAS performance. Link aggregation and redundant network configurations also ensure high availability.

A VLAN is a logical segment of a switched network or logical grouping of end devices connected to different physical networks. An end device could be a client or a NAS device. The segmentation or grouping can be done based on business functions, project teams, or applications. VLAN is a Layer 2 (data link layer) construct and works similar to a physical LAN. A network switch can be logically divided among multiple VLANs, enabling better utilization of the switch and reducing the overall cost of deploying a network infrastructure.

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

VLANs also provide enhanced security by restricting user access, flagging network intrusions, and controlling 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 setting for an Ethernet interface card is 1,500 bytes. A feature called *jumbo frames* sends, receives, or transports 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. However not all vendors use the same MTU size for jumbo frames. 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. Larger packets also reduce the amount of raw network bandwidth being consumed for the same amount of payload. Larger frames also help to smooth sudden I/O bursts.

The *TCP window size* is the maximum amount of data that can be sent 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 send only 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 overthe 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 or 65,536 bytes

The size of the 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 networkinterface, enabling higher throughput, loadsharing or load balancing, transparent path failover, and scalability. 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.

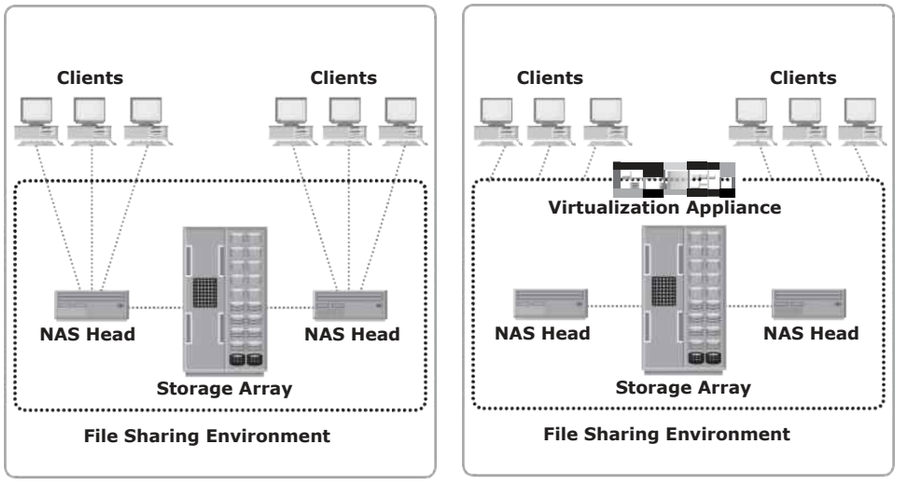
File-Level Virtualization

File-level virtualization eliminates the dependencies between the data accessed at the filelevel and the locationwhere the files are physicallystored. Implementation of file-level virtualization is common in NAS or file-server environments. It provides non-disruptive file mobility to optimize storage utilization.

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. Moving files across the environment is not easy and may make files inac- cessible during file movement. Moreover, hosts and applications need to be reconfigured to access the file at the new location. This makes it difficult for storage administrators to improve storage efficiency while maintaining the required service level.

File-level virtualization simplifies file mobility. It provides user or appli- cation independence from the location where the files are stored. File-level virtualization creates a logical pool of storage, enabling users to use a logi- cal path, rather than a physical path, to access files. File-level virtualization facilitates the movement of files across the online file servers or NAS devices. This means that while the files are being moved, clients can access their files nondisruptively. Clients can also read their files from the old location and write them back to the new location without realizing that the physical loca- tion has changed. A global namespace is used to map the logical path of a file to the physical path names.

- 7-9 illustrates a file-serving environment before and after the imple- mentation of file-level virtualization.



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

- 7-9: File-serving environment before and after file-level virtualization

Concepts in Practice: EMC Isilon and EMC VNX Gateway

EMC Isilon is the scale-out NAS solution. Isilon offers high scalability of both performance and storage capacity. It provides the capability to address big-data challenges.

The VNX Gateway, a member of the EMC VNX family, provides a gateway NAS solution. It provides multiprotocol file access, dynamic expansion of file systems, high availability, and high performance.

For more information on EMC Isilon and VNX Gateway, visit www.emc.com.

EMC Isilon

Isilon has a specialized operating system called OneFS that enables the scale- out NAS architecture. OneFS combines the three layers of traditional storage architectures — file system, volume manager, and RAID — into one unified software layer, creating a single file system that spans across all nodes in an Isilon cluster. OneFS enables data protection and automated data balancing. It provides the ability to seamlessly add storage and other resources without system downtime. With OneFS, throughput scales linearly with the number of nodes in a cluster.

OneFS enablesdifferent node types to be mixed in a single cluster through the addition of the SmartPools application software. SmartPools enables deploying a single file system to span multiple nodes that have different performance characteristics and capacities. Isilon offers different types of nodes, such as the X-Series, S-Series, NL-Series, and Accelerator. These nodes have different prices, performance levels, and storage capabilities. Each type of node is optimized for handling a specific type of workload.

OneFS enables the storage system administrator to specify the access pattern (random, concurrent, or sequential) on a per-file or per-directory basis. This unique capability enables OneFS to tailor data layout decisions, cache-retention

policies, and data prefetch policies to maximize performance of individual workflows.

OneFS constantly monitors the health of all files and disks within a cluster, and if components are at risk, the file system automatically flags the problem components for replacement and transparently relocates those files to healthy components.OneFSalsoensuresdataintegrityifthefilesystemhasanunexpected failure during a write operation.

When a new storage node is added, the Autobalance feature of OneFS automatically moves data onto this new node via the Infiniband based internal network. This automatic rebalancing ensures that the new node does not become a hot spot for new data. The Autobalance feature is transparent to the clients and can be adjusted to minimize the impact on high-performance workloads.

OneFS includes a core technology, called FlexProtect, to provide data protection. FlexProtect provides protection for up to four simultaneous failures of either nodes or individual drives per stripe. FlexProtect ensures minimal data reconstruction time if a failure occurs. FlexProtect provides file-specific protec- tion capabilities. Different protection levels can be assigned to individual files, directories, or to portions of a file system. These protection levels are aligned based on the importance of data and workflow.

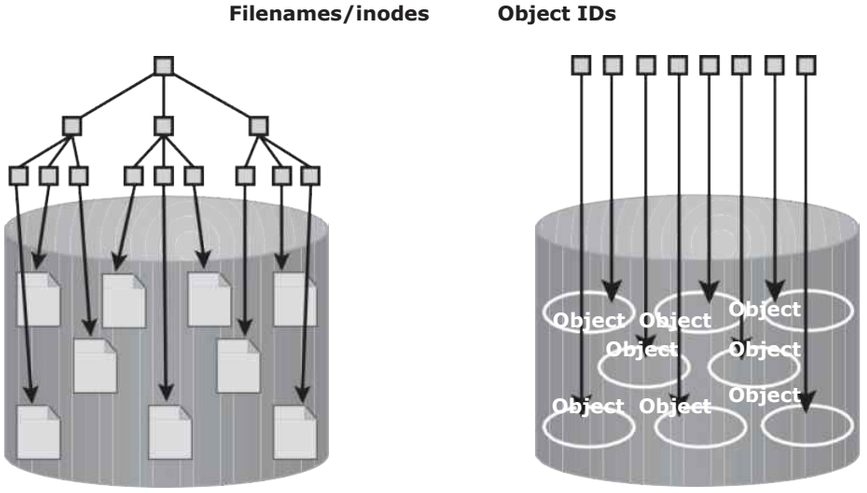
EMC VNX Gateway

The VNX Series Gateway contains one or more NAS heads, called X-Blades, that access external storage arrays, such as Symmetrix, block-based VNX, or CLARiiONstoragearray, via SAN. X-Bladesrun the VNX operating environment that is optimized for high-performance and multiprotocol network file system access. Each X-Blade consists of processors, redundant data paths, power sup- plies, Gigabit Ethernet, and 10-Gigabit Ethernet optical ports. All the X-Blades in a VNX gateway system are managed by Control Station, which provides a single point for configuring VNX Gateway. The VNX Gateway supports both pNFS and EMC patented Multi-Path File System(MPFS) protocols, which further improves the VNX Gateway performance.

VNX Series Gateway offers two models: VG2 and VG8. VG8 supports up to eight X-Blades, whereas VG2 supports up to two. X-Blades may be configured as either primary or standby. A primary X-Blade is the operating NAS head, whereas a standby X-Bladebecomes operational if the primary X-Bladefails. The Control Station handles an X-Blade failover. The Control Station also provides other high-availability features, such as fault monitoring, fault reporting, call home, and remote diagnostics.

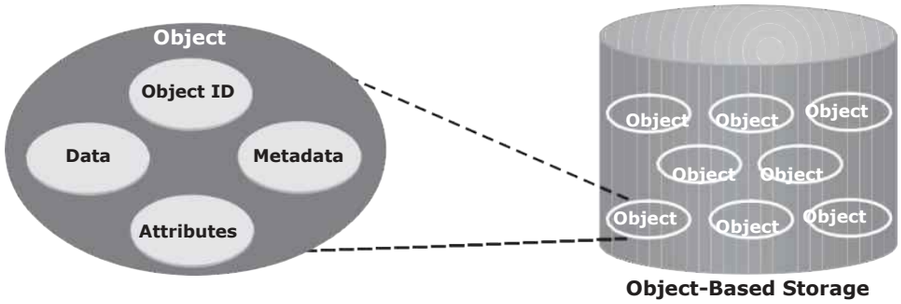
Object-Based Storage Devices

An OSD is a device that organizes and stores unstructured data, such as movies, office documents, and graphics, as objects. Object-based storage provides a scalable, self-managed, protected, and shared storage option. 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 - 8-1).



- 8-1: Hierarchical file system versus flat address space

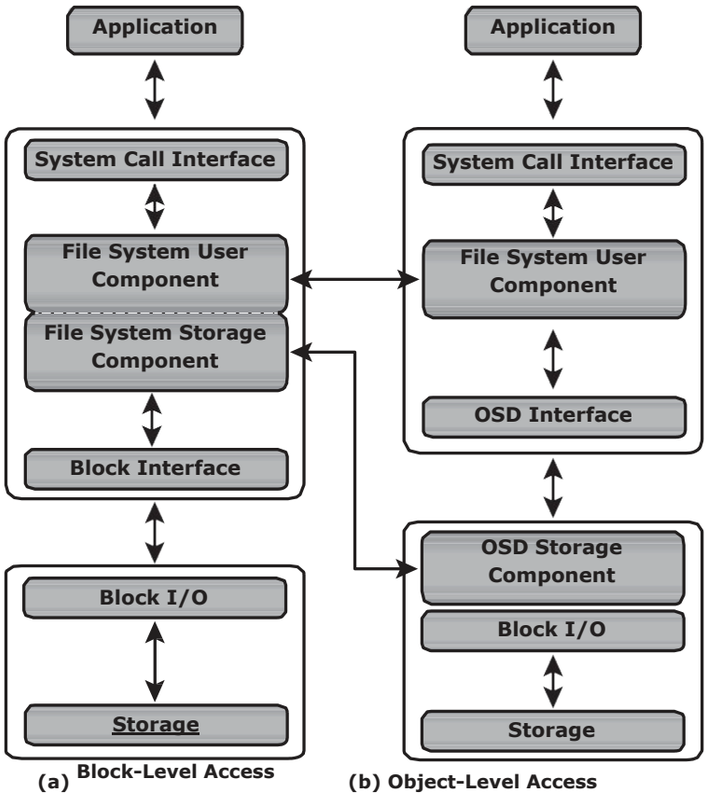
An object might contain user data, related metadata (size, date, ownership, and so on), and other attributes of data (retention, access pattern, and so on); see - 8-2. Each object stored in the system is identified by a unique ID called the *object ID*. The object ID is generated using specialized algorithms such as hash function on the dataand guarantees that everyobject is uniquely identified.



- 8-2: Object structure

Object-Based Storage Architecture

An I/O in the traditional block access method passes through various lay- ers in the I/O path. 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. - 8-3 (a) illustrates the block-level access.



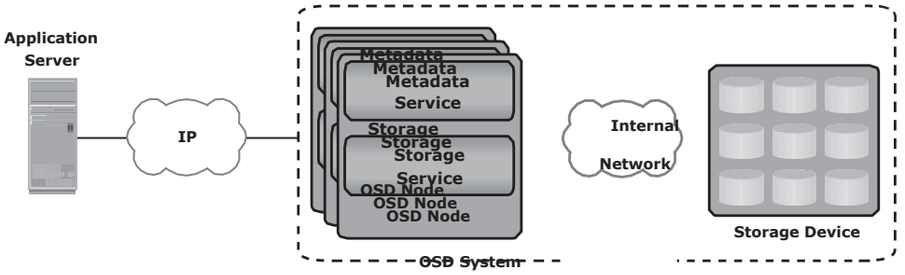
- 8-3: Block-level access versus object-level access

The file system has two components: user component and 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.

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. - 8-3 (b) illustrates the object-level access. 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.

Components of OSD

The OSD system is typically composed of three key components: nodes, private network, and storage. - 8-4 illustrates the components of OSD.



- 8-4: OSD components

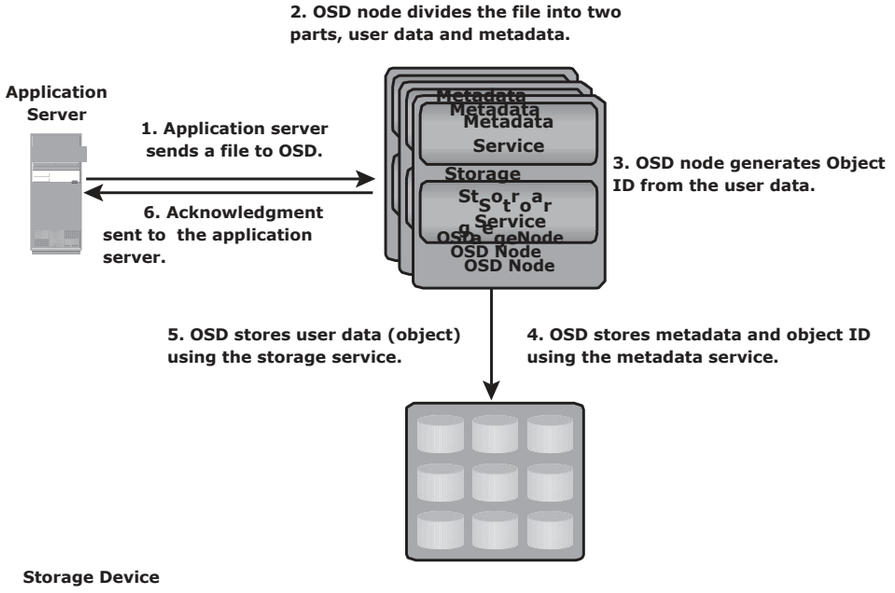
The OSD system is composed of one or more *nodes*. A node is a server that runs the OSD operating environment and provides services to store, retrieve, and manage data in the system. The OSD node has two key services: metadata service and storage service. The metadata service is responsible for generating the object ID from the contents (and can also include other attributes of data) 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. The application server accesses the node to store and retrieve data over an external network. In some implementations, such as CAS, the metadata service might reside on the application server or on a separate server.

OSD typically uses low-cost and high-density disk drives to store the objects. As more capacity is required, more disk drives can be added to the system.

Object Storage and Retrieval in OSD

The process of storing objects in OSD is illustrated in - 8-5. 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.
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.

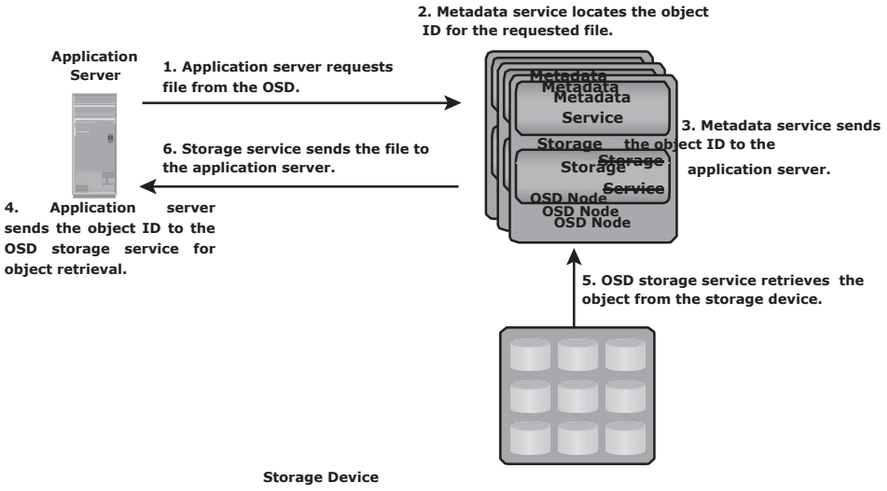


- 8-5: Storing objects on OSD

After an object is storedsuccessfully, it is available forretrieval. A user accesses the data stored on OSD by the same filename. The application server retrieves the stored content using the object ID. This process is transparent to the user.

The process of retrieving objects in OSD is illustrated in -s 8-6. The process of data retrieval from OSD is as follows:

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.
5. The OSD storage service retrieves the object from the storage device.
6. The OSD storage service sends the file to the application server.



- 8-6: Object retrieval from an OSD system

Benefits of Object-Based Storage

For unstructured data, object-based storage devices provide numerous benefits over traditional storage solutions. An ideal storage architecture should provide performance, scalability, security, and data sharing across multiple platforms. Traditional storage solutions, such as SAN and NAS, do not offer all these benefits as a single solution. Object-based storage combines benefits of both the worlds. It provides platform and location independence, and at the same time, provides scalability, security, and data-sharing capabilities. The key benefits of object-based storage are as follows:

- **Security and reliability:** Dataintegrity and content authenticity are the key features of object-based storage devices. OSD uses specialized algorithms

to create objects that provide strong data encryption capability. In OSD, request authentication is performed at the storage device rather than with an external authentication mechanism.

- **Platform independence:** Objects are abstract containers of data, including metadata and attributes. This feature allows objects to be shared across heterogeneousplatformslocally or remotely. Thisplatform-independence capability makesobject-basedstoragethebestcandidateforcloudcom- puting environments.
- **Scalability:** Duetotheuse of flataddressspace, object-based storagecan handlelargeamountsofdatawithoutimpactingperformance.Bothstorage and OSD nodes can be scaled independently in terms of performance and capacity.
- **Manageability:** Object-based storage has an inherent intelligencetomanage and protect objects. It uses self-healing capability to protect and replicate objects. Policy-based management capability helps OSD to handle routine jobs automatically.

Common Use Cases for Object-Based Storage

A data archival solution is a promising use case for OSD. Data integrity and protection is the primary requirement for any data archiving solu- tion. Traditional archival solutions — CD and DVD-ROM — do not provide scalability and performance. OSD stores data in the form of objects, associates them with a unique object ID, and ensures high data integrity. Along with integrity, it provides scalability and data protection. These capabilities make OSD a viable option for long term data archiving for fixed content. Content addressed storage (CAS) is a special type of object-based storage device purposely built for storing fixed content. CAS is covered in the fol- lowing section.

Another use case for OSD is cloud-based storage. OSD uses a web interface to access storage resources. OSD provides inherent security, scalability, and automated data management. It also enables data sharing across heterogeneous platforms or tenants while ensuring integrity of data. These capabilities make OSD astrong optionforcloud-basedstorage. Cloudservice providers can leverage OSD to offer storage-as-a-service.

OSD supports web service access via *representational state transfer* (REST) and *simple object access protocol* (SOAP). REST and SOAP APIs can be easily integrated with business applications that access OSD over the web.

Content-Addressed Storage

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. In CAS, the application server access the CAS device only via the CAS API running on the application server. However, the way CAS stores data is similar to the other OSD systems.

CAS provides all the features required for storing fixed content. The key features of CAS are as follows:

- **Content authenticity:** It assures the genuineness of stored content. This is achieved by generating a unique content address for each object and validating the content address for stored objects at regular intervals. Content authenticity is assured because the address assigned to each object is as unique as a fingerprint. Every time an object is read, CAS uses a hashing algorithm to recalculate the object's content address as a validation step and compares the result to its original content address. If the object fails validation, CAS rebuilds the object using a mirror or parity protection scheme.
- **Content integrity:** It provides assurance that the stored content has not been altered. CASusesahashingalgorithmforcontentauthenticityand integrity. If thefixedcontentisaltered, CAS generatesanewaddressfor the altered content, rather than overwrite the original fixed content.
- **Location independence:** CAS uses a unique content address, rather than directory path names or URLs, to retrieve data. This makes the physical location of the stored data irrelevant to the application that requests the data.
- **Single-instance storage (SIS):** CAS uses a unique content address to guar- antee the storage of only a single instance of an object. When a new object is written, the CAS system is polled to see whether an object is already available with the same content address. If the object is available in the system, it is not stored; instead, only a pointer to that object is created.
- **Retention enforcement:** Protecting and retaining objects is a core require- ment of an archive storage system. After an object is stored in the CAS systemandtheretentionpolicy isdefined, CASdoesnotmaketheobject available for deletion until the policyexpires.
- **Dataprotection:** CASensuresthatthecontentstoredontheCASsystemis available evenif a disk or a node fails. CAS provides both local and remote

protection to the data objects stored on it. In the local protection option, data objects are either mirrored or parity protected. In mirror protection, two copies of the data object are stored on two different nodes in the same cluster. This decreases the total available capacity by 50 percent. In parity protection, the data object is split in multiple parts and parity is generated from them. Each part of the data and its parity are stored on a different node. This method consumes less capacity to protect the stored data, but takes slightly longer to regenerate the data if corruption of data occurs.

In the remote replication option, data objects are copied to a secondary CAS at the remote location. In this case, the objects remain accessible from the secondary CAS if the primary CAS system fails.

- **Fast record retrieval:** CAS stores all objects on disks, which provides faster access to the objects compared to tapes and optical discs.
- **Loadbalancing:** CASdistributesobjectsacrossmultiplenodestoprovide maximum throughput and availability.
- **Scalability:** CAS allows the addition of more nodes to the cluster without anyinterruptiontodataaccessandwithminimumadministrativeoverhead.
- **Event notification:** CAS continuously monitors the state of the system and raises an alert for any event that requires the administrator's attention. The event notification is communicated to the administrator through SNMP, SMTP, or e-mail.
- **Self diagnosis and repair:** CAS automatically detects and repairs corruptedobjects and alerts theadministratoraboutthepotentialproblem. CAS systems can be configured to alert remote support teams who can diagnose and repair the system remotely.
- **Audit trails:** CAS keeps track of management activities and any access or dispositionofdata.Audittrailsaremandatedbycompliancerequirements.

CAS Use Cases

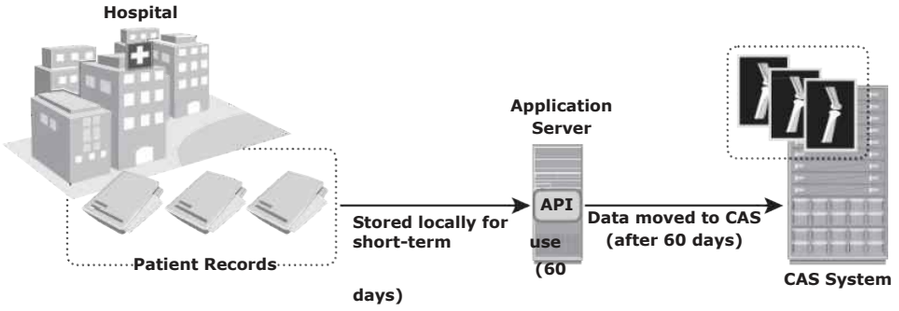
Organizations have deployed CAS solutions to solve several business challenges. Two solutions are described in detail in the following sections.

Healthcare Solution: Storing Patient Studies

Large healthcare centers examine hundreds of patients every day and generate large volumes of medical records. Each record might be composed of one

or more images that range in size from approximately 15 MB for a standard digital X-ray to more than 1 GB for oncology studies. The patient records are stored online for a specific period of time for immediate use by the attending physicians. Even if a patient's record is no longer needed, compliance requirements might stipulate that the records be kept in the original format for several years.

Medicalimage solutionprovidersoffer hospitalsthe capabilitytoview medical records, such as X-ray images, with acceptable response times and resolution to enable rapid assessments of patients. - 8-7 illustrates the use of 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.

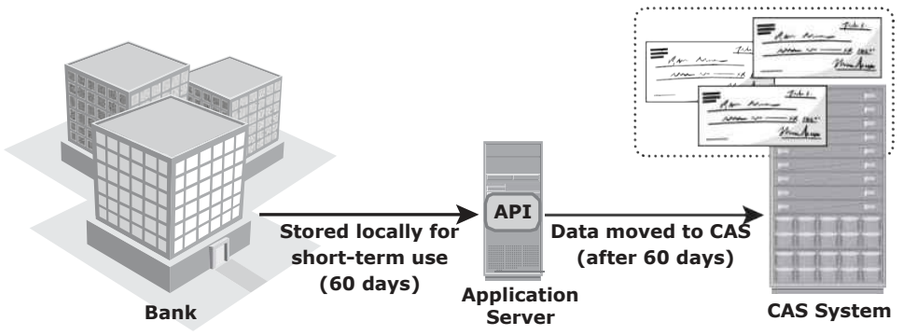


- 8-7: Storing patient studies on a CAS system

Finance Solution: Storing Financial Records

In a typical banking scenario, images of checks, each approximately 25 KB in size, are created and sent to archive services over an IP network. A check imaging service provider might process approximately 90 million check images per month. Typically, check images are actively processed in transactional systems for about 5 days.

For the next 60 days, check images may be requested by banks or individual consumers for verification purposes; beyond 60 days, access requirements drop drastically. - 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. Check imaging is one example of a financial service application that is best serviced with CAS. Customer transactions initiated by e-mail, contracts, and security transaction records might need to be kept online for 30 years; CAS is the preferred storage solution in such cases.



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

Unified Storage

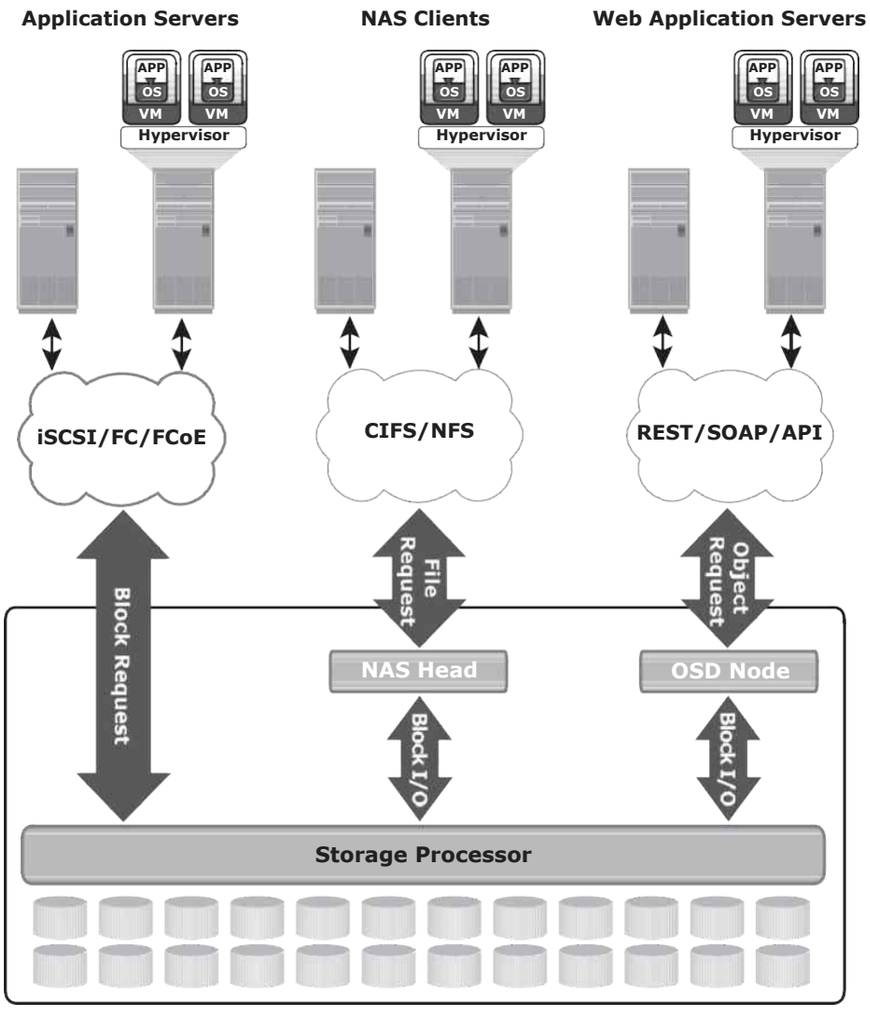
Unified storage consolidates block, file, and object access into one storage solu- tion. It supports multiple protocols, such as CIFS, NFS, iSCSI, FC, FCoE, REST (representational state transfer), and SOAP (simple object access protocol).

Components of Unified Storage

A unified storage system consists of the following key components: storage controller, NAS head, OSD node, and storage. - 8-9 illustrates the block diagram of a unified storage platform.

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. The controller configures LUNs and presents them to application servers, NAS heads, and OSD nodes. The LUNs presented to the application server appear as local physical disks. A file system is configured on these LUNs and is made available to applications for storing data.

A *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. The system typically has two or more NAS heads for redundancy. The LUNs presented to the NAS head appear as physical disks. The NAS head configures the file systems on these disks, creates a NFS, CIFS, or mixed share, and exports the share to the NAS clients.



- 8-9: Unified storage platform

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.

Data Access from Unified Storage

In a unified storage system, block, file, and object requests to the storage travel through different I/O paths. - 8-9 illustrates the different I/O paths for block, file, and object access.

- **Block I/Orequest:** Theapplication serversareconnectedtoan FC, iSCSI, or FCoE port on the storage controller. The server sends a block request over an FC, iSCSI, or FCoE connection. The storage processor (SP) pro- cesses the I/O and responds to the application server.
- **File I/O request:** The NAS clients (where the NAS share is mounted or mapped) send a file request to the NAS head using the NFS or CIFS pro- tocol. The NAS head receives the request, converts it into a block request, and forwards it to the storage controller. Upon receiving the block data from the storage controller, the NAS head again converts the block request back to the file request and sends it to the clients.
- **Object I/O request:** The web application servers send an object request, typically using REST or SOAP protocols, to the OSD node. The OSD node receives the request, converts it into a block request, and sends it to the disk through the storage controller. The controller in turn processes the block request and responds back to the OSD node, which in turn provides the requested object to the web application server.

Concepts in Practice: EMC Atmos, EMC VNX, and EMC Centera

EMCAtmossupportsobject-basedstoragefor unstructured data, such as pictures and videos. Atmos combines massive scalability with specialized intelligence to address the cost, distribution, and management challenges associated with vast amounts of unstructured data.

EMC VNX is a unified storage platform that consolidates block, file, and object access inone solution. It implementsamodular architecture that integrates hard- ware components for block, file, and object access. EMC VNX delivers file access (NAS) functionality via X-Blades (Data Movers) and block access functionality via storage processors. Optionally, it offers object access to the storage using EMC Atmos Virtual Edition (Atmos VE).

EMC Centera is a simple, affordable, and secure repository for information archiving. EMC Centera is designed and optimized specifically to deal with the storage and retrieval of fixed content by meeting performance, compliance, and regulatory requirements. Compared to traditional archive storage, EMC Centera provides faster record retrieval, Single instance storage (SIS), guaran- teed content authenticity, self-healing, and support for numerous industry and regulatory standards.

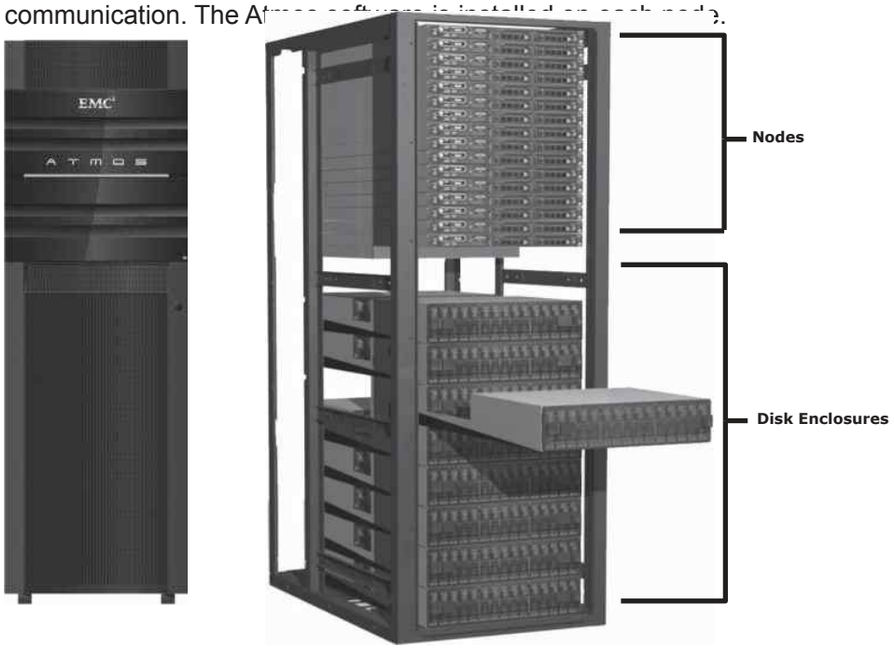
For the latest information on EMC Atmos, EMC VNX, and EMC Centera, visit www.emc.com.

EMC Atmos

Atmos can be deployed in two ways: as a purpose-built hardware appliance or as software in VMware environments, where AtmosVE can leverage the existing servers and storage.

- 8-10 illustrates the EMC Atmos hardware appliance. The hardware appliance iscomprised of servers (nodes) connected tostandard disk enclosures.

The rack includes a 24-port Gigabit Ethernet switch to provide internode communication. The At[illegible].



- 8-10: EMC Atmos storage system

Atmos VE enables users to exploit the power of Atmos in a virtualized environment. It can be deployed on a virtual machine in VMware ESXi hosts and configured with the VMware certified back-end storage.

Following are the key features offered by EMC Atmos:

- **Policy-based management:** EMC Atmos improves operational efficiency by automatically distributing content based on business policy. The administrator-defined policies dictate how, when, and where the information resides.

- **Protection:** Atmos offers two options to protect the objects, replication and Geo Parity:
  - *Replication* ensures that the content is available and accessible by creating redundant copies of an object at redundant designated locations.
  - *GeoParity* ensures that the content is available and accessible by dividing objects into multiple segments plus parity segments and distributing them to one or more designated locations.
- **Data services:** EMC Atmos includes the data services, such as compression and deduplication. These features are native to Atmos and can be man- aged and accessed via a policy.
- **Web services and legacy protocols:** EMC Atmos provides flexible web services access (REST/SOAP) for web-scale applications and file access (CIFS/NFS/Installable File System/Centera API) for traditional applications.
- **Automated system management:** EMC Atmos provides auto-configuring, auto- managing, and auto-healing capabilities to reduce administration and downtime.
- **Multitenancy:** EMC Atmos enables multiple applications to be served from the same infrastructure. Each application is securely partitioned and cannot access the other application's data. Multitenancy is ideal for service providers or large enterprises that want to provide cloud computing services to multiple customers or departments allowing logical and secure separation within a single infrastructure.
- **Flexible administration:** EMC Atmos can be managed via a graphical user interface (GUI) or command-line interface (CLI).

EMC VNX

VNX is EMC's unified storage product offering. - 8-11 illustrates the EMC VNX storage array.

VNX storage systems include the following components:

- *Storage processors* (SPs) support block I/O access to storage with FC, iSCSI, and FCoE protocols.
- *X-Blades* access data from the back end and provide host access with NFS, CIFS, MPFS, pNFS, and FTP protocols. The X-Blades in each array are scalable and provide redundancy to ensure no single point of failure.
- *Control Stations* provide management functions to the X-Blades. The Control Station is also responsible for X-Blade failover. The Control Station may optionally be configured with a matching secondary Control Station to ensure management redundancy on the VNX array.

- *Standby power supplies* provide enough power to each storage processor and first DAE to ensure that any data in flight is stored in the vault area if a power failure occurs. This ensures that no writes are lost.
- *Disk-array enclosures* (DAEs) house the drives used in the array. Different sized DAEs are available that can each hold a maximum of 15, 25, or 60 drives. More DAEs can be added to meet growing storage demands.

- 8-11: EMC VNX storage system

EMC Centera

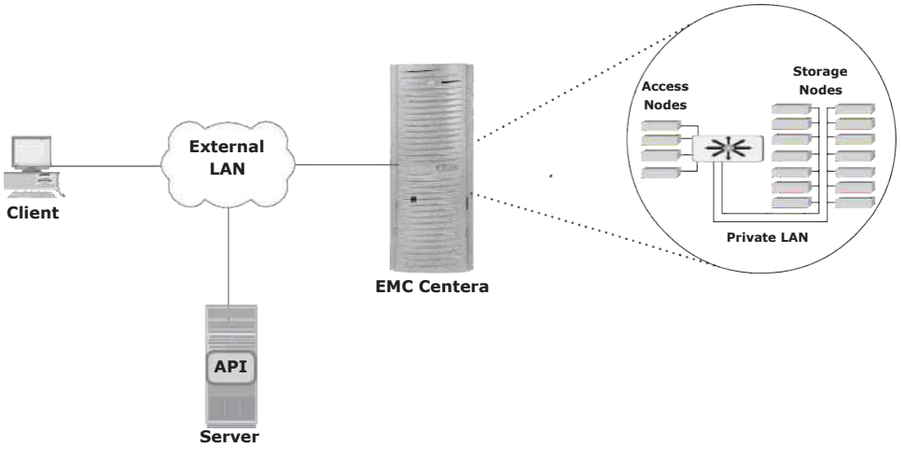
EMC Centera is offered in three different models to meet different types of user requirements — EMC Centera Basic, EMC Centera Governance Edition, and EMC Centera Compliance Edition Plus (CE+):

- **EMC Centera Basic:** Provides all functionalities without the enforcement of retention periods.

- **EMC Centera Governance Edition:** Provides the retention capabilities required by organizations to manage digital records in addition to the features provided by EMC Centera Basic.
- **EMC Centera Compliance Edition Plus:** Provides extensive compliance capabilities. CE+ is designed to meet the requirements of the most stringent regulated business environments for electronic storagemedia, as established by regulations from the Securities and Exchange Commission (SEC), or other national and international regulatory groups.

EMC Centera Architecture

The Centera architecture is shown in - 8-12. A client accesses the Centera over a LAN. The client can access Centera only through the server that runs the Centera API (application programming interface). The Centera API is responsible for performing functions that enable an application to store and retrieve the data.



- 8-12: Centera architecture

Centera architecture is a *Redundant Array of Independent Nodes* (RAIN). It con- tains storage nodes and access nodes that are networked as a cluster by using a private LAN. The internal LAN reconfigures automatically when it detects configuration changes, such as the addition of storage or access nodes. The application server accesses the Centera via an external LAN.

Thenodes are configured withlow-cost, high-capacity SATAdiskdrives. These nodes run CentraStar, the operating environment for Centera, which provides the features and functionalities required in a Centera system.

When nodes are installed, they are configured with a —roleǁ that defines the functionality provided to the node. A node can be configured as a storage node, an access node, or a dual-role node.

Storage nodes store and protect data objects. They are sometimes referred to as *back-end nodes*.

Access nodes provide connectivity to application servers through an external LAN. They establish connectivity with the storage nodes in the cluster through a private LAN. The number of access nodes is determined by the amount of throughput required from the cluster. If a node is configured solely as an —access node,ǁ its disk space cannot be used to store data objects. Storage and retrieval requests are sent to the access node via the external LAN.

Dual-role nodes provide both storage and access-node capabilities. This configuration is more common than a pure access-node configuration.