DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING
INTERNAL ASSESSMENT TEST – 3
STORAGE AREA NETWORKS (10IS765)
QUESTION PAPER & KEY

USN □□□□□□□□□□

Improvement Test – Nov. 2017

Sub:	STORAGE AREA NETWORKS (SAN)					Sub Code:	10CS765	Branch: CSE
Date:	18.11.2017	Duration:	90 min's	Max Marks:	50	Sem / Sec:	VII	OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1 (a)	Consider a disk I/O system, I/O request arrives at a rate of 100 I/Os per second. The service time Rs is 8ms. Compute the following measures of disk performance: a. Utilization of I/O controller, b. Total response time, c. Average Queue size, & d. Total time spent by request in a queue. Now, if controller power is doubled, the service time is halved; consequently, Rs = 4ms. In this scenario, determine the above measures.	[10]	CO1	L3
2 (a)	With a neat diagram, discuss the features of high end storage systems.	[10]	C02	L3
3 (a)	Explain the different FC ports with a neat diagram.	[10]	C03	L4
4 (a)	What is storage virtualization? Describe the types of storage virtualization in detail with diagram.	[10]	C04	L3
5 (a)	With neat diagram, explain the steps involved in BC backup and restore operation .	[10]	C05	L3
6 (a)	Describe SCSI - 3 architecture in detail with diagram.	[10]	C03	L4
7 (a)	Describe the failure analysis in BC. Briefly explain BC technology solution.	[10]	C05	L3

1. Consider a disk I/O system, I/O request arrives at a rate of 100 I/Os per second. The service time Rs is 8ms. Compute the following measures of disk performance:

a. Utilization of I/O controller,
b. Total response time,
c. Average Queue size, &
d. Total time spent by request in a queue.

Now, if controller power is doubled, the service time is halved; consequently, Rs = 4ms. In this scenario, determine the above measures.

Solution:

- Arrival Rate (a)

R_a= 1/a = 1 sec / 100 = 1000ms/100

R_a = 10 ms

- Rs = 8 ms (Given)

a) Utilization (U) = Rs / Ra = 8 / 10 = 0.8 or 80%
b) Response Time (R) = Rs / (1-U) = 8 / (1 – 0.8) = 8 / 0.2 = 40 ms
c) Average Queue Size = U / (1-U) = 0.8 / (1-0.8) = 0.8 / 0.2 = 4
d) Total time spent by request in a queue = U * R = 0.8 * 40 ms = 32 ms

- Arrival Rate (a)

R_a= 1/a = 1 sec / 200 = 1000ms/200

R_a = 5 ms

- Rs =4 ms (Given)

a) Utilization (U) = Rs / Ra = 4 / 5 = 0.8 or 80%
b) Response Time (R) = Rs / (1-U) = 4 / (1 – 0.8) = 4 / 0.2 = 20 ms
c) Average Queue Size = U / (1-U) = 0.8 / (1-0.8) = 0.8 / 0.2 = 4
d) Total time spent by request in a queue = U * R = 0.8 * 20 ms = 16 ms

2. With a neat diagram, discuss the features of high end storage systems.
Solution:

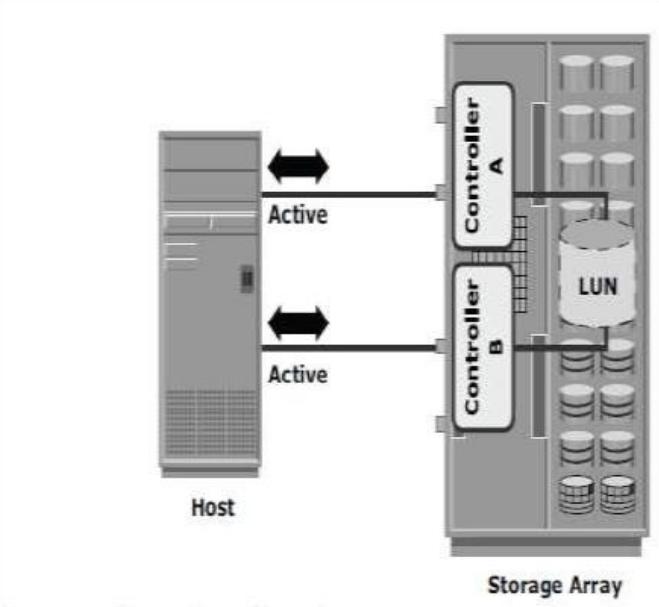


Figure 4-7: Active-active configuration

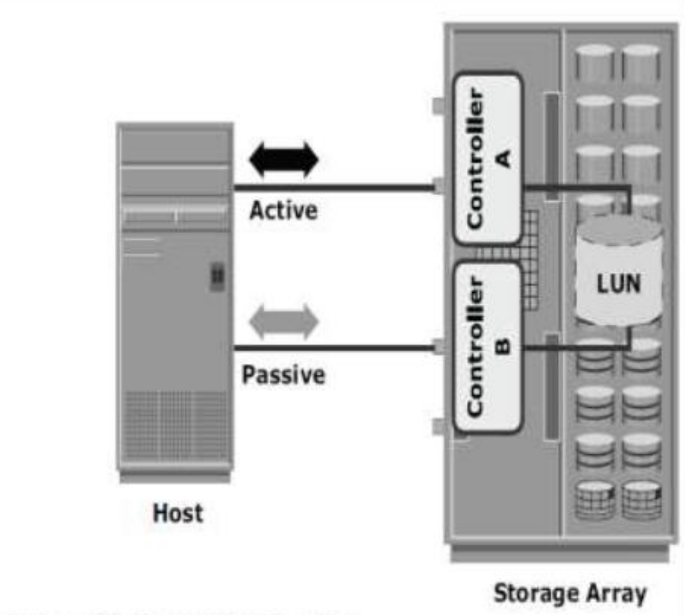


Figure 4-8: Active-passive configuration

High-end storage systems, referred to as ***active-active arrays***, are aimed at large enterprises for centralizing corporate data. These arrays are designed with a large number of controllers and cache memory. An active-active array implies that the host can perform I/Os to its LUNs across any of the available paths (see Figure 4-7).

To address the enterprise storage needs, these arrays provide the following capabilities:
1. Large storage capacity
2. Large amounts of cache to service host I/Os optimally
3. Fault tolerance architecture to improve data availability
4. Connectivity to mainframe computers and open systems hosts
5. Availability of multiple front-end ports and interface protocols to serve a large number of hosts
6. Availability of multiple back-end Fibre Channel or SCSI RAID controllers to manage disk processing
7. Scalability to support increased connectivity, performance, and storage capacity requirements
8. Ability to handle large amounts of concurrent I/Os from a number of servers and applications
9. Support for array-based local and remote replication

3. Explain the different FC ports with a neat diagram.
Solution:

Fibre Channel Ports

Ports are the basic building blocks of an FC network. Ports on the switch can be one of the following types:

1. 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 that is connected to a switch in a switched fabric.

2. NL_port: A node port that supports the arbitrated loop topology. This port is also known as the *node loop port*.

3. E_port: An FC port that forms the connection between two FC switches. This port is also known as the *expansion port*. The E_port on an FC switch connects to the E_port of another FC switch in the fabric through a link, which is called an *Inter-Switch Link (ISL)*. ISLs are used to transfer host-to-storage data as well as the fabric management traffic from one switch to another. ISL is also one of the scaling mechanisms in SAN connectivity.

4. F_port: A port on a switch that connects an N_port. It is also known as a *fabric port* and cannot participate in FC-AL.

5. FL_port: A fabric port that participates in FC-AL. This port is connected to the NL_ports on an FC-AL loop. A FL_port also connects a loop to a switch in a switched fabric. As a result, all NL_ports in the loop can participate in FC-SW. This configuration is referred to as a *public loop*. In contrast, an arbitrated loop without any switches is referred to as a *private loop*. A private loop contains nodes with NL_ports, and does not contain FL_port.

6. G_port: A generic port that can operate as an E_port or an F_port and determines its functionality automatically during initialization.

Figure 6-12 shows various FC ports located in the fabric.

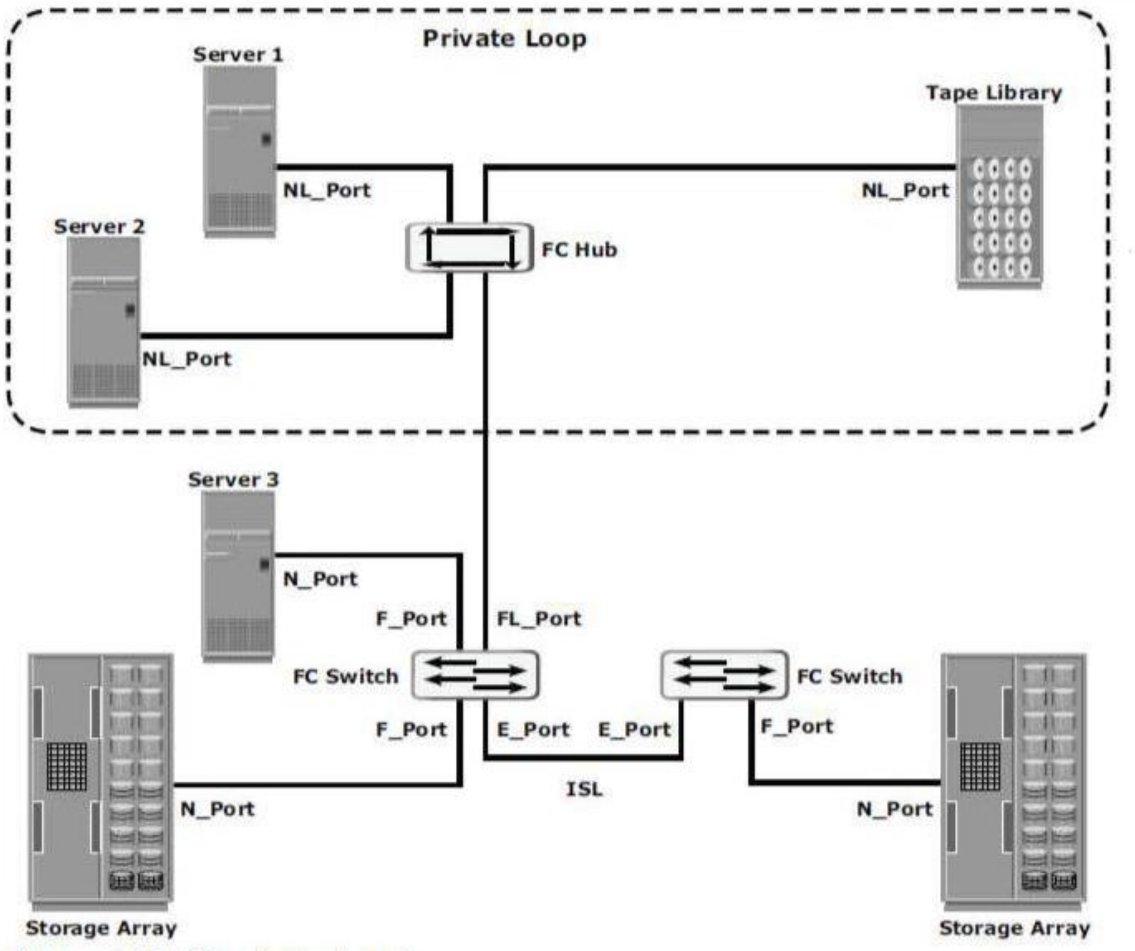


Figure 6-12: Fibre channel ports

4. What is storage virtualization? Describe the types of storage virtualization in detail with diagram.

Solution:

Virtual storage is about providing logical storage to hosts and applications independent of physical resources.

Virtualization can be implemented in both SAN and NAS storage environments.

In a SAN, virtualization is applied at the block level, whereas in NAS, it is applied at the file level.

10.5.1 Block-Level Storage Virtualization

- *Block-level storage virtualization* provides a translation layer in the SAN, between the hosts and the storage arrays, as shown in Figure 10-6.
- Instead of being directed to the LUNs on the individual storage arrays, the hosts are directed to the virtualized LUNs on the virtualization device.
- The virtualization device translates between the virtual LUNs and the physical LUNs on the individual arrays. This facilitates the use of arrays from different vendors simultaneously, without any interoperability issues.
- For a host, all the arrays appear like a single target device and LUNs can be distributed or even split across multiple arrays.
- Block-level storage virtualization extends storage volumes online, resolves application growth requirements, consolidates heterogeneous storage arrays, and enables transparent volume access. It also provides the advantage of non-disruptive data migration.
- In traditional SAN environments, LUN migration from one array to another was an offline event because the hosts needed to be updated to reflect the new array configuration.
- With a block-level virtualization solution in place, the virtualization engine handles the back-end migration of data, which enables LUNs to remain online and accessible while data is being migrated. No physical changes are required because the host still points to the same virtual targets on the virtualization device.

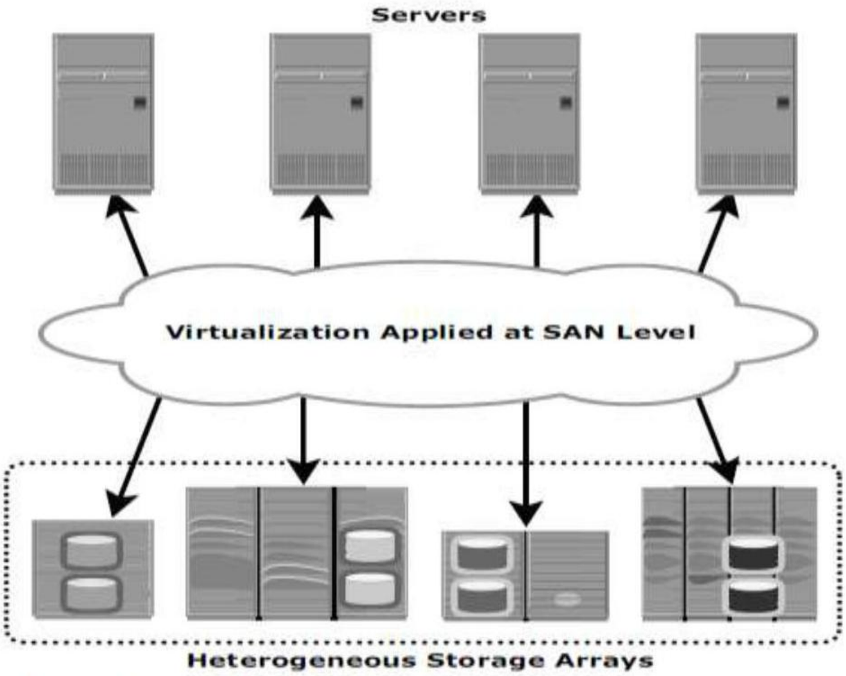

Servers
Virtualization Applied at SAN Level
Heterogeneous Storage Arrays

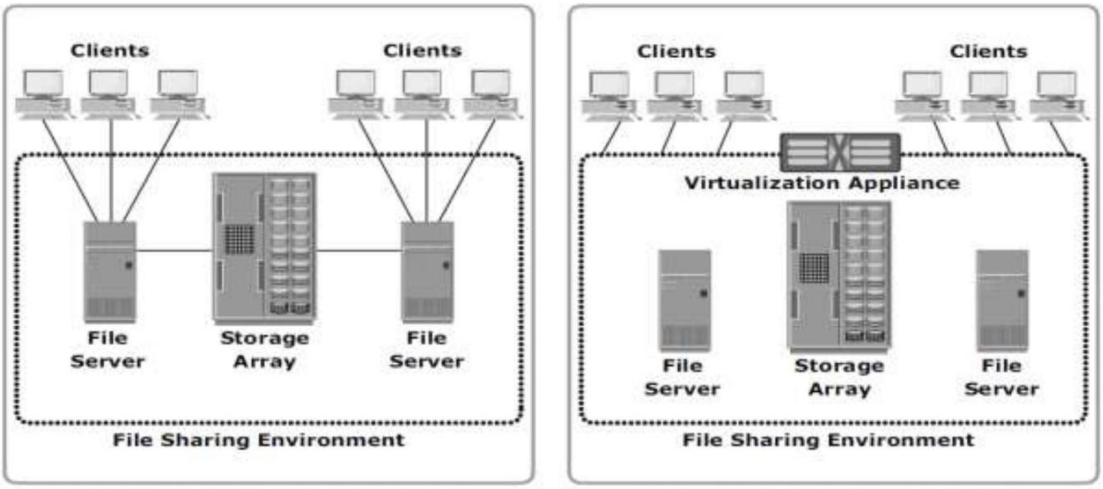
Figure 10-6: Block-level storage virtualization

10.5.2 File-Level Virtualization

- *File-level virtualization* addresses the NAS challenges by eliminating the dependencies between the data accessed at the file level and the location where the files are physically stored.
- This provides opportunities to optimize storage utilization and server consolidation and to perform nondisruptive file migrations. Figure 10-7 illustrates a NAS environment before and after the implementation of file-level virtualization.
- Before virtualization, each NAS device or file server is physically and logically independent. Each host knows exactly where its file-level resources are located. Underutilized storage resources and capacity problems result because files are bound to a specific file server.
- It is necessary 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 requires downtime for the file servers.
- Moreover, hosts and applications need to be reconfigured with the new path, making 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 application independence from the location where the files are stored.
- File-level virtualization creates a logical pool of storage, enabling users to use a logical path, rather than a physical path, to access files.
- File-level virtualization facilitates the movement of file systems across the online file servers. This means that while the files are being moved, clients can access their files non-disruptively.

- Clients can also read their files from the old location and write them back to the new location without realizing that the physical location has changed.
- Multiple clients connected to multiple servers can perform online movement of their files to optimize utilization of their resources. A global namespace can be used to map the logical path of a file to the physical path names.



(a) Before File-Level Virtualization

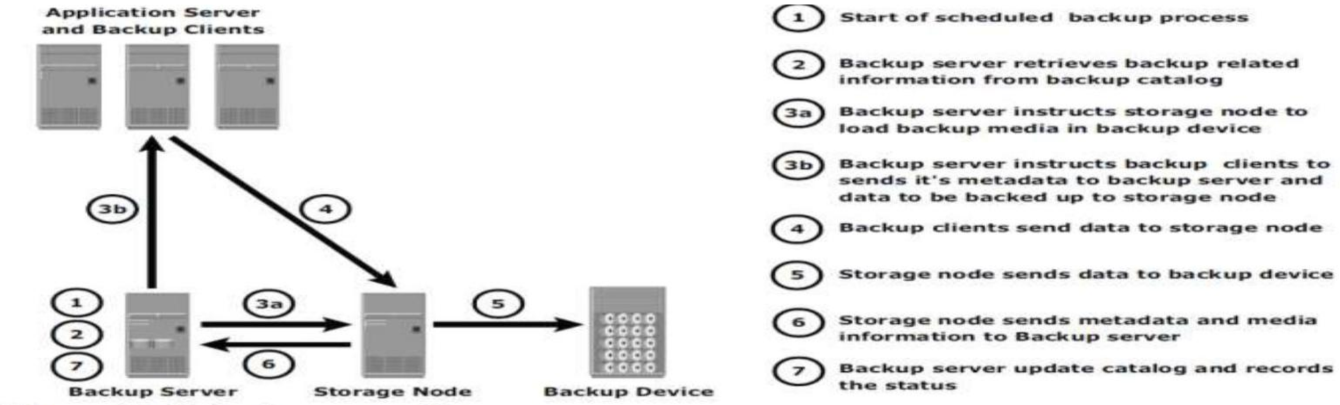
(b) After File-Level Virtualization

Figure 10-7: NAS device before and after file-level virtualization

5. With neat diagram, explain the steps involved in BC backup and restore operation . Solution:

When a backup process is initiated, significant network communication takes place between the different components of a backup infrastructure. The backup server initiates the backup process for different clients based on the backup schedule configured for them. For example, the backup process for a group of clients may be scheduled to start at 3:00 am every day.

The backup server coordinates the backup process with all the components in a backup configuration (see Figure 12-5). The backup server maintains the information about backup clients to be contacted and storage nodes to be used in a backup operation. The backup server retrieves the backup-related information from the backup catalog and, based on this information, instructs the storage node to load the appropriate backup media into the backup devices. Simultaneously, it instructs the backup clients to start scanning the data, package it, and send it over the network to the assigned storage node. The storage node, in turn, sends metadata to the backup server to keep it updated about the media being used in the backup process. The backup server continuously updates the backup catalog with this information.



1 Start of scheduled backup process

2 Backup server retrieves backup related information from backup catalog

3a Backup server instructs storage node to load backup media in backup device

3b Backup server instructs backup clients to sends it's metadata to backup server and data to be backed up to storage node

4 Backup clients send data to storage node

5 Storage node sends data to backup device

6 Storage node sends metadata and media information to Backup server

7 Backup server update catalog and records the status

Figure 12-5: Backup operation

After the data is backed up, it can be restored when required. A restore process must be manually initiated. Some backup software has a separate application for restore operations. These restore applications are accessible only to the administrators.

Figure 12-6 depicts a restore process.

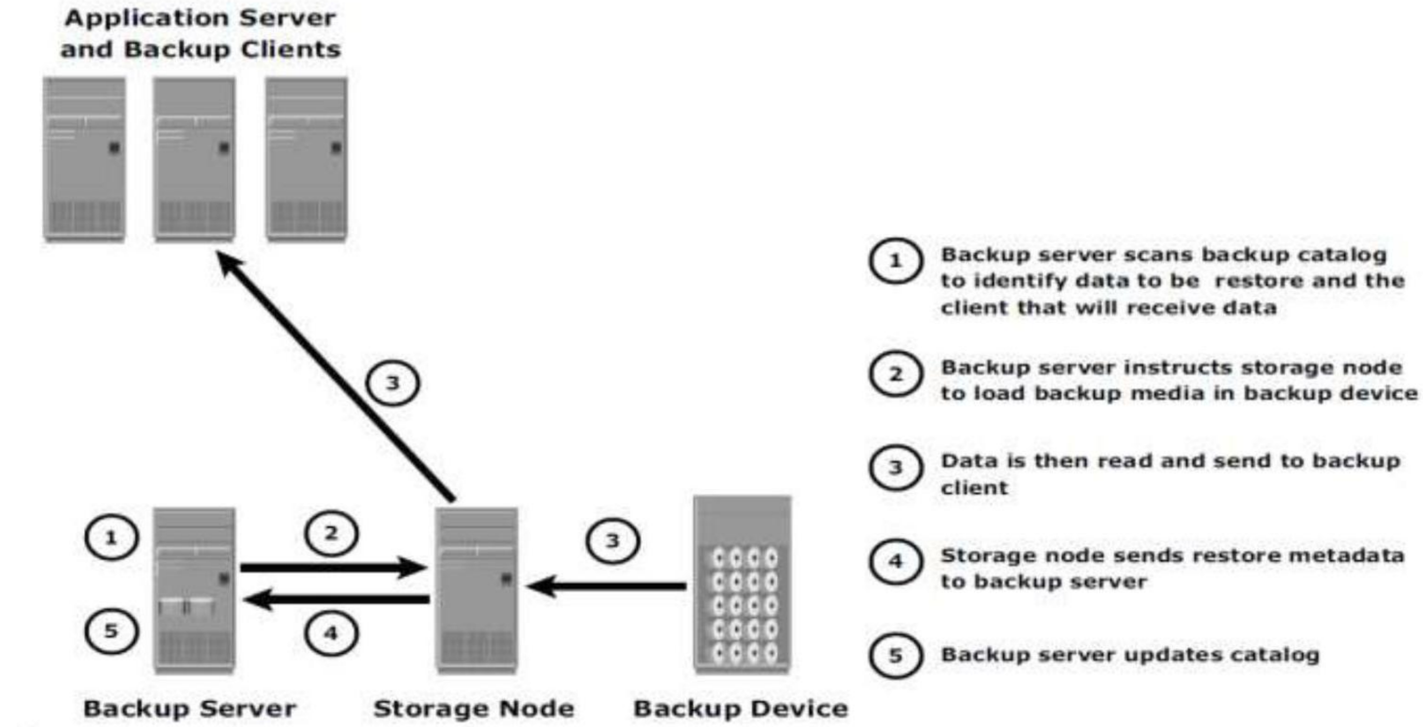

Application Server and Backup Clients
3
1
2
3
5
4
Backup Server
Storage Node
Backup Device
1 Backup server scans backup catalog to identify data to be restore and the client that will receive data
2 Backup server instructs storage node to load backup media in backup device
3 Data is then read and send to backup client
4 Storage node sends restore metadata to backup server
5 Backup server updates catalog

Figure 12-6: Restore operation

Upon receiving a restore request, an administrator opens the restore application to view the list of clients that have been backed up. While selecting the client for which a restore request has been made, the administrator also needs to identify the client that will receive the restored data. Data can be restored on the same client for whom the restore request has been made or on any other client.

The administrator then selects the data to be restored and the specified point in time to which the data has to be restored based on the RPO. Note that because all of this information comes from the backup catalog, the restore application must also communicate to the backup server.

The administrator first selects the data to be restored and initiates the restore process. The backup server, using the appropriate storage node, then identifies the backup media that needs to be mounted on the backup devices. Data is then read and sent to the client that has been identified to receive the restored data.

6. Describe SCSI - 3 architecture in detail with diagram.

Solution:

The SCSI-3 architecture defines and categorizes various SCSI-3 standards and requirements for SCSI-3 implementations. The SCSI-3 architecture was approved and published as the standard X.3.270-1996 by the ANSI. This architecture helps developers, hardware designers, and users to understand and effectively utilize SCSI.

The three major components of a SCSI architectural model are as follows:

1. SCSI-3 command protocol: This consists of primary commands that are common to all devices as well as device-specific commands that are unique to a given class of devices.

2. Transport layer protocols: These are a standard set of rules by which devices communicate and share information.

3. Physical layer interconnects: These are interface details such as electrical signaling methods and data transfer modes.

Common access methods are the ANSI software interfaces for SCSI devices.

Figure 5-3 shows the SCSI-3 standards architecture with interrelated groups of other standards within SCSI-3.

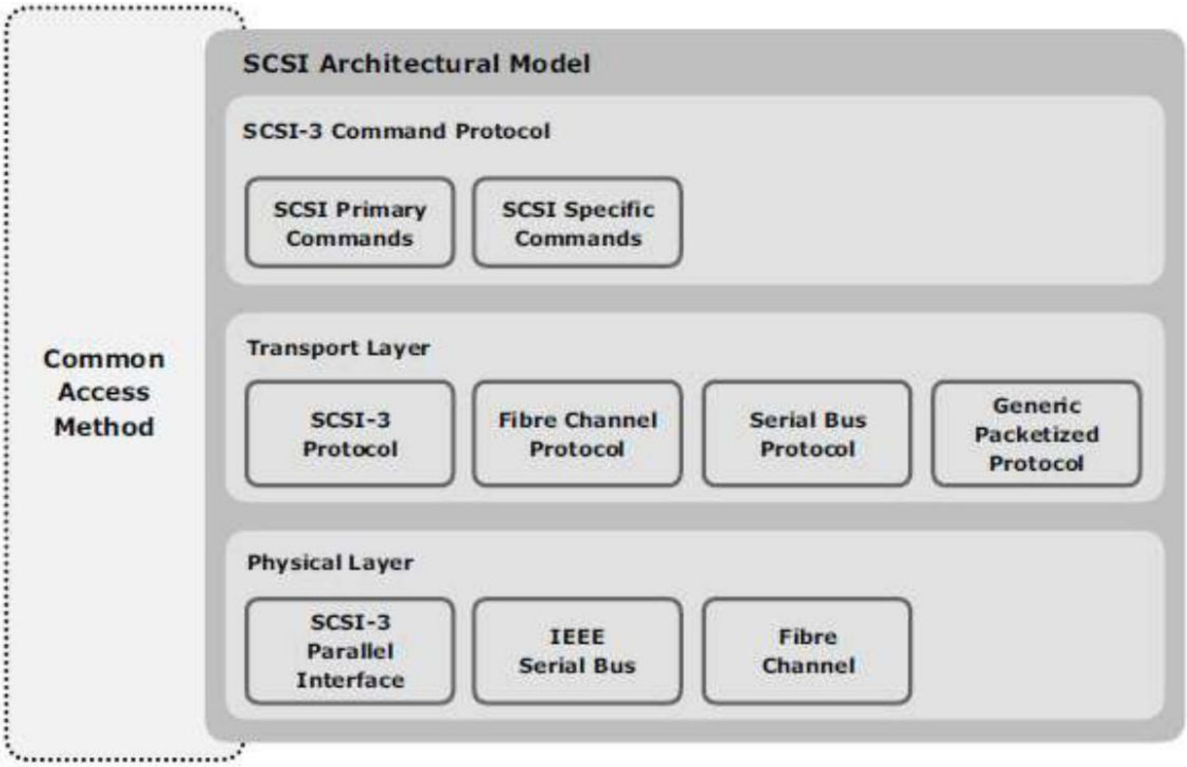


Figure 5-3: SCSI-3 standards architecture

7(a) Describe the failure analysis in BC. Briefly explain BC technology solution.
Solution:

Failure analysis involves analyzing the data center to identify systems that are susceptible to a single point of failure and implementing fault-tolerance mechanisms such as redundancy.

11.4.1 Single Point of Failure

A single point of failure refers to the failure of a component that can terminate the availability of the entire system or IT service. Figure 11-4 illustrates the possibility of a single point of failure in a system with various components: server, network, switch, and storage array. The figure depicts a system setup in which an application running on the server provides an interface to the client and performs I/O operations. The client is connected to the server through an IP network, the server is connected to the storage array through a FC connection, an HBA installed at the server sends or receives data to and from a storage array, and an FC switch connects the HBA to the storage port.

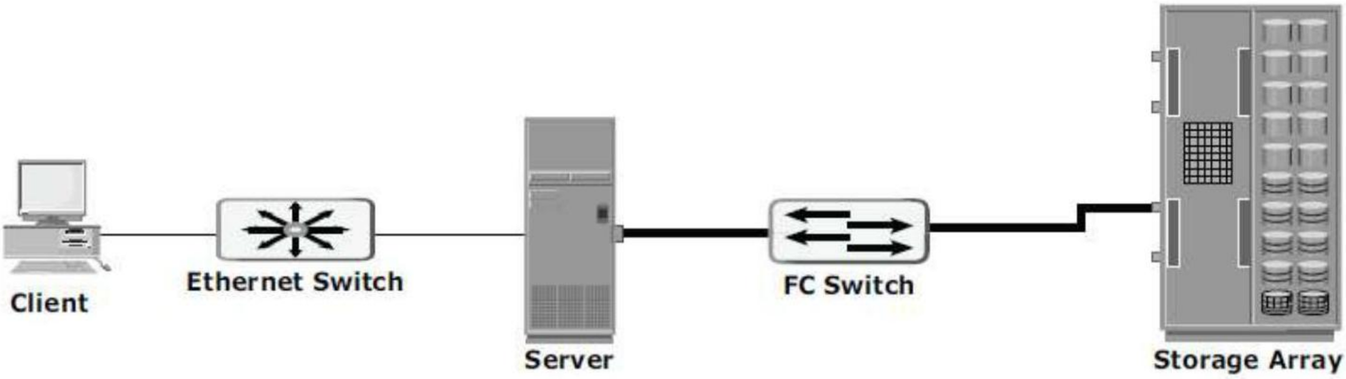


Figure 11-4: Single point of failure

In a setup where each component must function as required to ensure data availability, the failure of a single component causes the failure of the entire data center or an application, resulting in disruption of business operations. In this example, several single points of failure can be identified. The single HBA on the server, the server itself, the IP network, the FC switch, the storage array ports, or even the storage array could become potential single points of failure. To avoid single points of failure, it is essential to implement a fault-tolerant mechanism.

11.4.2 Fault Tolerance

To mitigate a single point of failure, systems are designed with redundancy, such that the system will fail only if all the components in the redundancy group fail. This ensures that the failure of a single component does not affect data availability. Figure 11-5 illustrates the fault-tolerant implementation of the system just described (and shown in Figure 11-4).

Data centers follow stringent guidelines to implement fault tolerance. Careful analysis is performed to eliminate every single point of failure. In the example shown in Figure 11-5, all enhancements in the infrastructure to mitigate single points of failures are emphasized:

1. Configuration of multiple HBAs to mitigate single HBA failure.
2. Configuration of multiple fabrics to account for a switch failure.
3. Configuration of multiple storage array ports to enhance the storage array’s availability.
4. RAID configuration to ensure continuous operation in the event of disk failure.
5. Implementing a storage array at a remote site to mitigate local site failure.
6. Implementing server (host) clustering, a fault-tolerance mechanism whereby two or more servers in a cluster access the same set of volumes. Clustered servers exchange *heartbeats* to inform each other about their health.

If one of the servers fails, the other server takes up the complete workload.

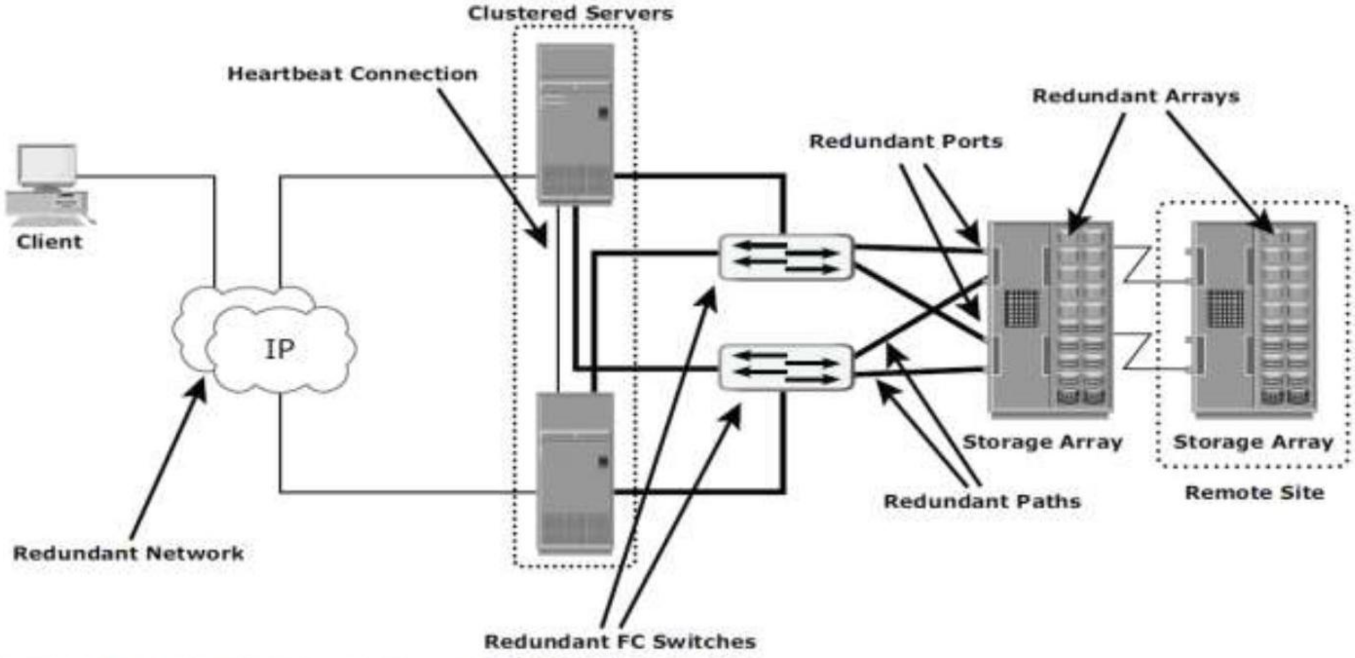


Figure 11-5: Implementation of fault tolerance

11.4.3 Multipathing Software

Configuration of multiple paths increases the data availability through path failover. If servers are configured with one I/O path to the data there will be no access to the data if that path fails. Redundant paths eliminate the path to become single points of failure. Multiple paths to data also improve I/O performance through load sharing and maximize server, storage, and data path utilization.

In practice, merely configuring multiple paths does not serve the purpose. Even with multiple paths, if one path fails, I/O will not reroute unless the system recognizes that it has an alternate path. Multipathing software provides the functionality to recognize and utilize alternate I/O path to data. Multipathing software also manages the load balancing by distributing I/Os to all available, active paths.