Chapter 12
Backup and Recovery

A *backup* is a copy of production data, created and retained for the sole purpose of recovering deleted or corrupted data. With growing business and regulatory demands for data storage, retention, and availability, organizations are faced with the task of backing up an ever-increasing amount of data. This task becomes more challenging as demand for consistent backup and quick restore of data increases throughout the enterprise — which may be spread over multiple sites. Moreover, organizations need to accomplish backup at a lower cost with minimum resources.

KEY CONCEPTS

- Operational Backup
- Archival
- Retention Period
- Bare-Metal Recovery
- Backup Architecture
- Backup Topologies
- Virtual Tape Library

Organizations must ensure that the right data is in the right place at the right time. Evaluating backup technologies, recovery, and retention requirements for data and applications is an essential step to ensure successful implementation of the backup and recovery solution. The solution must facilitate easy recovery and retrieval from backups and archives as required by the business.

This chapter includes details about the purposes of backup, strategies for backup and recovery operations, backup methods, the backup architecture, and backup media.

12.1 Backup Purpose

Backups are performed to serve three purposes: disaster recovery, operational backup, and archival.

12.1.1 Disaster Recovery

Backups can be performed to address disaster recovery needs. The backup copies are used for restoring data at an alternate site when the primary site is incapacitated due to a disaster. Based on RPO and RTO requirements, organizations use different backup strategies for disaster recovery. When a tape-based backup method is used as a disaster recovery strategy, the backup tape media is shipped and stored at an offsite location. These tapes can be recalled for restoration at the disaster recovery site. Organizations with stringent RPO and RTO requirements use remote replication technology to replicate data to a disaster recovery site. This allows organizations to bring up production systems online in a relatively short period of time in the event of a disaster. Remote replication is covered in detail in Chapter 14.

12.1.2 Operational Backup

Data in the production environment changes with every business transaction and operation. *Operational backup* is a backup of data at a point in time and is used to restore data in the event of data loss or logical corruptions that may occur during routine processing. The majority of restore requests in most organizations fall in this category. For example, it is common for a user to accidentally delete an important e-mail or for a file to become corrupted, which can be restored from operational backup.

Operational backups are created for the active production information by using incremental or differential backup techniques, detailed later in this chapter. An example of an operational backup is a backup performed for a production database just before a bulk batch update. This ensures the availability of a clean copy of the production database if the batch update corrupts the production database.

12.1.3 Archival

Backups are also performed to address archival requirements. Although CAS has emerged as the primary solution for archives, traditional backups are still used by small and medium enterprises for long-term preservation of transaction

records, e-mail messages, and other business records required for regulatory compliance.

Apart from addressing disaster recovery, archival, and operational requirements, backups serve as a protection against data loss due to physical damage of a storage device, software failures, or virus attacks. Backups can also be used to protect against accidents such as a deletion or intentional data destruction.

12.2 Backup Considerations

The amount of data loss and downtime that a business can endure in terms of RTO and RPO are the primary considerations in selecting and implementing a specific backup strategy. Another consideration is the retention period, which defines the duration for which a business needs to retain the backup copies. Some data is retained for years and some only for a few days. For example, data backed up for archival is retained for a longer period than data backed up for operational recovery.

It is also important to consider the backup media type, based on the retention period and data accessibility. Organizations must also consider the granularity of backups, explained later in this chapter. The development of a backup strategy must include a decision about the most appropriate time for performing a backup in order to minimize any disruption to production operations. Similarly, the location and time of the restore operation must be considered, along with file characteristics and data compression that influences the backup process.

Location, size, and number of files should also be considered, as they may affect the backup process. Location is an important consideration for the data to be backed up. Many organizations have dozens of heterogeneous platforms supporting complex solutions. Consider a data warehouse environment that uses backup data from many sources. The backup process must address these sources in terms of transactional and content integrity. This process must be coordinated with all heterogeneous platforms on which the data resides.

File size also influences the backup process. Backing up large-size files (example: ten 1 MB files) may use less system resources than backing up an equal amount of data comprising a large number of small-size files (example: ten thousand 1 KB files). The backup and restore operation takes more time when a file system contains many small files.

Like file size, the number of files to be backed up also influences the backup process. For example, in incremental backup, a file system containing one million files with a 10 percent daily change rate will have to create 100,000 entries in the backup catalog, which contains the table of contents for the backed up data set

and information about the backup session. This large number of entries in the file system affects the performance of the backup and restore process because it takes a long time to search through a file system.

Backup performance also depends on the media used for the backup. The time-consuming operation of starting and stopping in a tape-based system affects backup performance, especially while backing up a large number of small files.

Data compression is widely used in backup systems because compression saves space on the media. Many backup devices, such as tape drives, have built-in support for hardware-based data compression. To effectively use this, it is important to understand the characteristics of the data. Some data, such as application binaries, do not compress well. Text data does compress well, whereas other data such as JPEG and ZIP files are already compressed.

12.3 Backup Granularity

Backup granularity depends on business needs and required RTO/RPO. Based on granularity, backups can be categorized as full, cumulative, and incremental. Most organizations use a combination of these three backup types to meet their backup and recovery requirements. Figure 12-1 depicts the categories of backup granularity.

Full backup is a backup of the complete data on the production volumes at a certain point in time. A full backup copy is created by copying the data on the production volumes to a secondary storage device. *Incremental backup* copies the data that has changed since the last full or incremental backup, whichever has occurred more recently. This is much faster (because the volume of data backed up is restricted to changed data), but it takes longer to restore. *Cumulative* (or *differential*) *backup* copies the data that has changed since the last full backup. This method takes longer than incremental backup but is faster to restore.

Synthetic (or *constructed*) *full backup* is another type of backup that is used in implementations where the production volume resources cannot be exclusively reserved for a backup process for extended periods to perform a full backup. It is usually created from the most recent full backup and all the incremental backups performed after that full backup. A synthetic full backup enables a full backup copy to be created offline without disrupting the I/O operation on the production volume. This also frees up network resources from the backup process, making them available for other production uses.

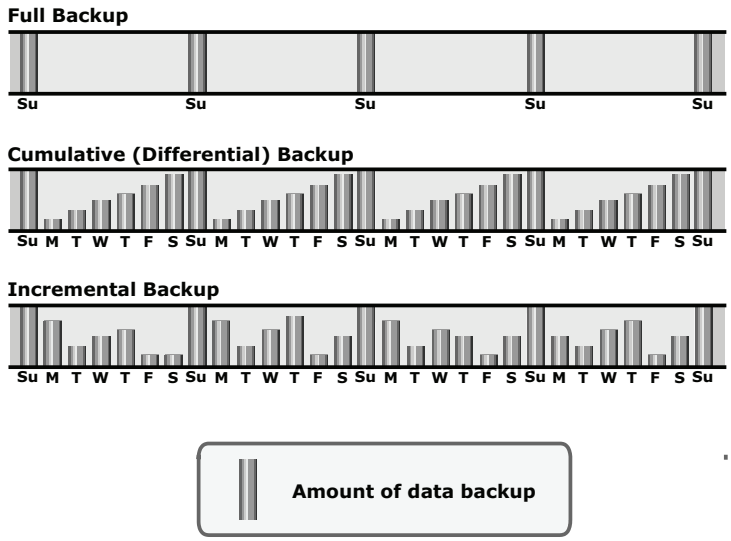

Figure 12-1: Backup granularity levels

Restore operations vary with the granularity of the backup. A full backup provides a single repository from which data can be easily restored. The process of restoration from an incremental backup requires the last full backup and all the incremental backups available until the point of restoration. A restore from a cumulative backup requires the last full backup and the most recent cumulative backup. Figure 12-2 illustrates an example of an incremental backup and restoration.

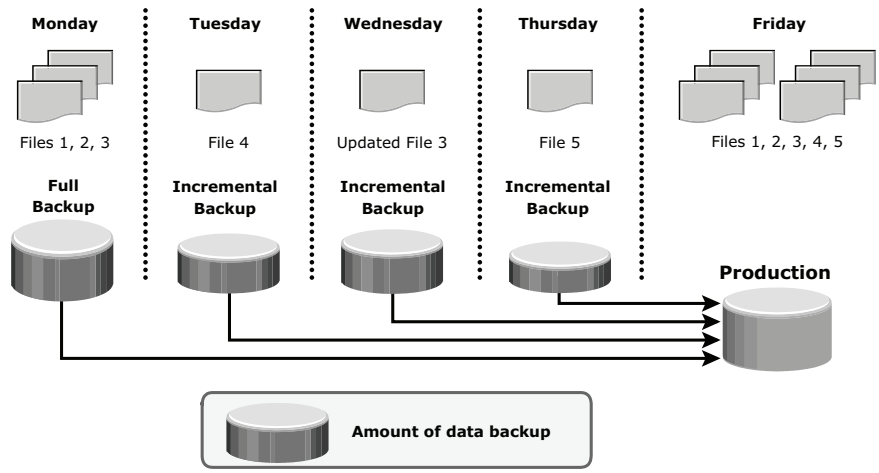

Figure 12-2: Restoring from an incremental backup

In this example, a full backup is performed on Monday evening. Each day after that, an incremental backup is performed. On Tuesday, a new file (File 4 in the figure) is added, and no other files have changed. Consequently, only File 4 is copied during the incremental backup performed on Tuesday evening. On Wednesday, no new files are added, but File 3 has been modified. Therefore, only the modified File 3 is copied during the incremental backup on Wednesday evening. Similarly, the incremental backup on Thursday copies only File 5. On Friday morning, there is data corruption, which requires data restoration from the backup. The first step toward data restoration is restoring all data from the full backup of Monday evening. The next step is applying the incremental backups of Tuesday, Wednesday, and Thursday. In this manner, data can be successfully restored to its previous state, as it existed on Thursday evening. Figure 12-3 illustrates an example of cumulative backup and restoration.

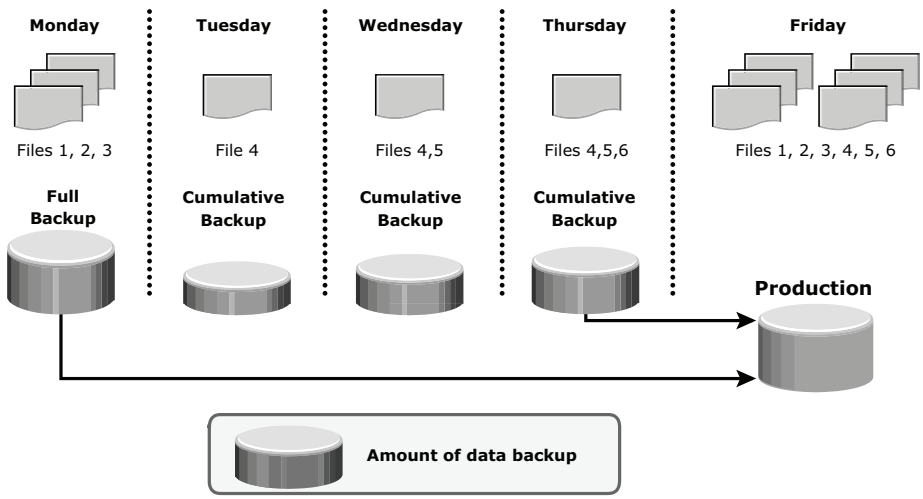

Figure 12-3: Restoring a cumulative backup

In this example, a full backup of the business data is taken on Monday evening. Each day after that, a cumulative backup is taken. On Tuesday, File 4 is added and no other data is modified since the previous full backup of Monday evening. Consequently, the cumulative backup on Tuesday evening copies only File 4. On Wednesday, File 5 is added. The cumulative backup taking place on Wednesday evening copies both File 4 and File 5 because these files have been added or modified since the last full backup. Similarly, on Thursday, File 6 is added. Therefore, the cumulative backup on Thursday evening copies all three files: File 4, File 5, and File 6.

On Friday morning, data corruption occurs that requires data restoration using backup copies. The first step in restoring data from a cumulative backup is restoring all data from the full backup of Monday evening. The next step is to apply only the latest cumulative backup — Thursday evening. In this way, the production volume data can be easily restored to its previous state on Thursday evening.

12.4 Recovery Considerations

RPO and RTO are major considerations when planning a backup strategy. RPO defines the tolerable limit of data loss for a business and specifies the time interval between two backups. In other words, the RPO determines backup frequency. For example, if application A requires an RPO of one day, it would need the data to be backed up at least once every day.

The retention period for a backup is also derived from an RPO specified for operational recovery. For example, users of application "A" may request to restore the application data from its operational backup copy, which was created a month ago. This determines the retention period for the backup. The RPO for application A can therefore range from one day to one month based on operational recovery needs. However, the organization may choose to retain the backup for a longer period of time because of internal policies or external factors, such as regulatory directives.

If short retention periods are specified for backups, it may not be possible to recover all the data needed for the requested recovery point, as some data may be older than the retention period. Long retention periods can be defined for all backups, making it possible to meet any RPO within the defined retention periods. However, this requires a large storage space, which translates into higher cost. Therefore, it is important to define the retention period based on an analysis of all the restore requests in the past and the allocated budget.

RTO relates to the time taken by the recovery process. To meet the defined RTO, the business may choose to use a combination of different backup solutions to minimize recovery time. In a backup environment, RTO influences the type of backup media that should be used. For example, recovery from data streams multiplexed in tape takes longer to complete than recovery from tapes with no multiplexing.

Organizations perform more full backups than they actually need because of recovery constraints. Cumulative and incremental backups depend on a previous full backup. When restoring from tape media, several tapes are needed to fully recover the system. With a full backup, recovery can be achieved with a lower RTO and fewer tapes.

12.5 Backup Methods

Hot backup and cold backup are the two methods deployed for backup. They are based on the state of the application when the backup is performed. In a *hot backup*, the application is up and running, with users accessing their data during the backup process. In a *cold backup*, the application is not active during the backup process.

The backup of online *production data* becomes more challenging because data is actively being used and changed. An open file is locked by the operating system and is not copied during the backup process until the user closes it. The backup application can back up open files by retrying the operation on files that were opened earlier in the backup process. During the backup process, it may be possible that files opened earlier will be closed and a retry will be successful. The maximum number of retries can be configured depending on the backup application. However, this method is not considered robust because in some environments certain files are always open.

In such situations, the backup application provides *open file agents*. These agents interact directly with the operating system and enable the creation of consistent copies of open files. In some environments, the use of open file agents is not enough. For example, a database is composed of many files of varying sizes, occupying several file systems. To ensure a consistent database backup, all files need to be backed up in the same state. That does not necessarily mean that all files need to be backed up at the same time, but they all must be synchronized so that the database can be restored with consistency.

Consistent backups of databases can also be done by using a cold backup. This requires the database to remain inactive during the backup. Of course, the disadvantage of a cold backup is that the database is inaccessible to users during the backup process.

Hot backup is used in situations where it is not possible to shut down the database. This is facilitated by *database backup agents* that can perform a backup while the database is active. The disadvantage associated with a hot backup is that the agents usually affect overall application performance.

A *point-in-time (PIT)* copy method is deployed in environments where the impact of downtime from a cold backup or the performance resulting from a hot backup is unacceptable. A pointer-based PIT copy consumes only a fraction of the storage space and can be created very quickly. A pointer-based PIT copy is implemented in a disk-based solution whereby a virtual LUN is created and holds pointers to the data stored on the production LUN or save location. In this method of backup, the database is stopped or frozen momentarily while the PIT copy is created. The PIT copy is then mounted on a secondary server and the backup occurs on the primary server. This technique is detailed in Chapter 13.

To ensure consistency, it is not enough to back up only production data for recovery. Certain attributes and properties attached to a file, such as permissions,

owner, and other metadata, also need to be backed up. These attributes are as important as the data itself and must be backed up for consistency. Backup of boot sector and partition layout information is also critical for successful recovery.

In a disaster recovery environment, *bare-metal recovery (BMR)* refers to a backup in which all metadata, system information, and application configurations are appropriately backed up for a full system recovery. BMR builds the base system, which includes partitioning, the file system layout, the operating system, the applications, and all the relevant configurations. BMR recovers the base system first, before starting the recovery of data files. Some BMR technologies can recover a server onto dissimilar hardware.

12.6 Backup Process

A backup system uses client/server architecture with a backup server and multiple backup clients. The backup server manages the backup operations and maintains the backup catalog, which contains information about the backup process and backup metadata. The backup server depends on backup clients to gather the data to be backed up. The backup clients can be local to the server or they can reside on another server, presumably to back up the data visible to that server. The backup server receives backup metadata from the backup clients to perform its activities.

Figure 12-4 illustrates the backup process. The storage node is responsible for writing data to the backup device (in a backup environment, a storage node is a host that controls backup devices). Typically, the storage node is integrated with the backup server and both are hosted on the same physical platform. A backup device is attached directly to the storage node's host platform. Some backup architecture refers to the storage node as the *media server* because it connects to the storage device. Storage nodes play an important role in backup planning because they can be used to consolidate backup servers.

The backup process is based on the policies defined on the backup server, such as the time of day or completion of an event. The backup server then initiates the process by sending a request to a backup client (backups can also be initiated by a client). This request instructs the backup client to send its metadata to the backup server, and the data to be backed up to the appropriate storage node. On receiving this request, the backup client sends the metadata to the backup server. The backup server writes this metadata on its metadata catalog. The backup client also sends the data to the storage node, and the storage node writes the data to the storage device.

After all the data is backed up, the storage node closes the connection to the backup device. The backup server writes backup completion status to the metadata catalog.

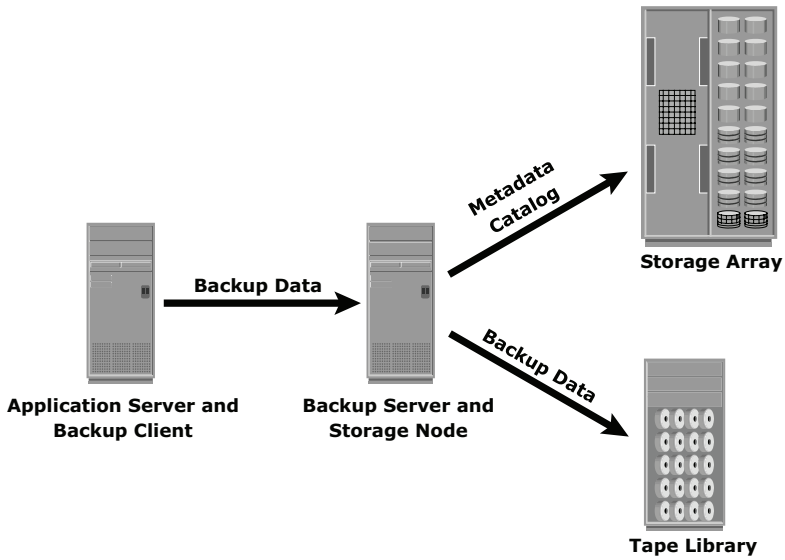

Figure 12-4: Backup architecture and process

Backup software also provides extensive reporting capabilities based on the backup catalog and the log files. These reports can include information such as the amount of data backed up, the number of completed backups, the number of incomplete backups, and the types of errors that may have occurred. Reports can be customized depending on the specific backup software used.

12.7 Backup and Restore Operations

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.

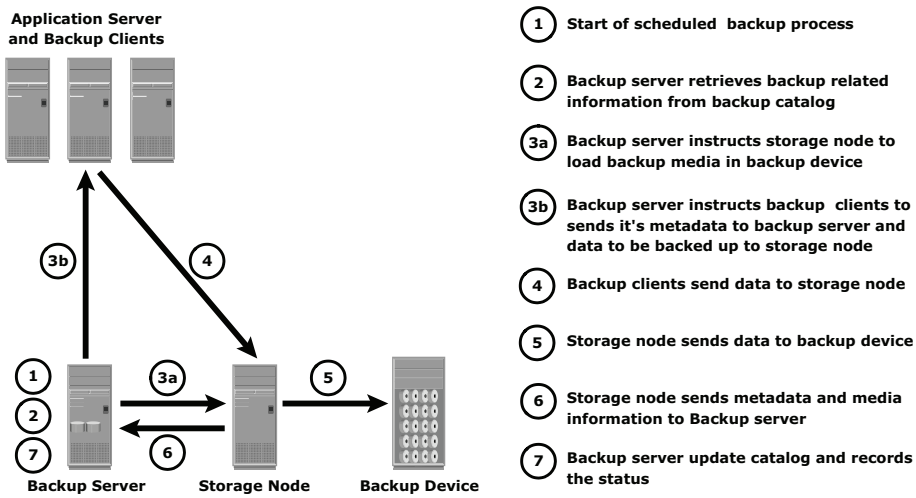

Figure 12-5: Backup operation

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.

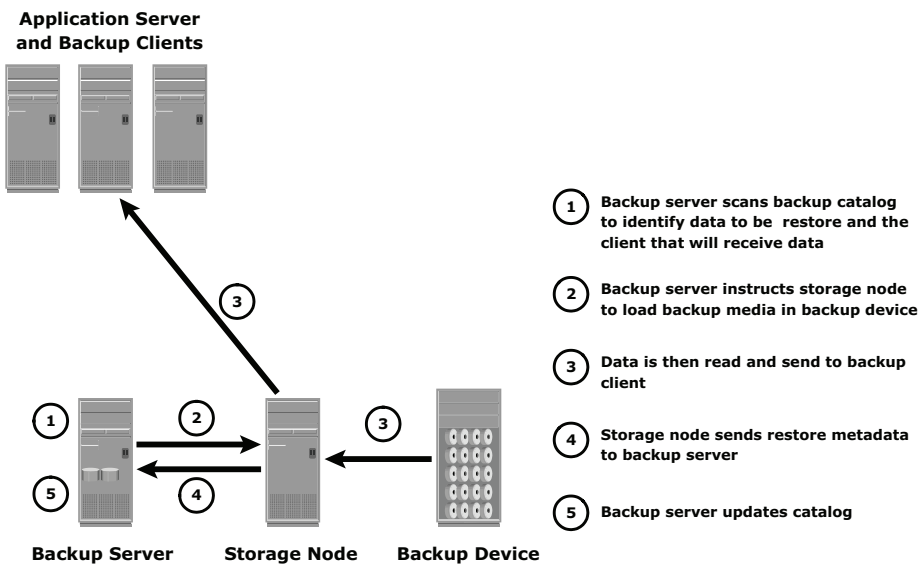

Figure 12-6: Restore operation

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.

Some restorations are successfully accomplished by recovering only the requested production data. For example, the recovery process of a spreadsheet is completed when the specific file is restored. In database restorations, additional data such as log files and production data must be restored. This ensures application consistency for the restored data. In these cases, the RTO is extended due to the additional steps in the restoration process.

12.8 Backup Topologies

Three basic topologies are used in a backup environment: direct attached backup, LAN based backup, and SAN based backup. A mixed topology is also used by combining LAN based and SAN based topologies.

In a *direct-attached backup*, a backup device is attached directly to the client. Only the metadata is sent to the backup server through the LAN. This configuration frees the LAN from backup traffic. The example shown in Figure 12-7 depicts use of a backup device that is not shared. As the environment grows, however, there will be a need for central management of all backup devices and to share the resources to optimize costs. An appropriate solution is to share the backup devices among multiple servers. In this example, the client also acts as a storage node that writes data on the backup device.

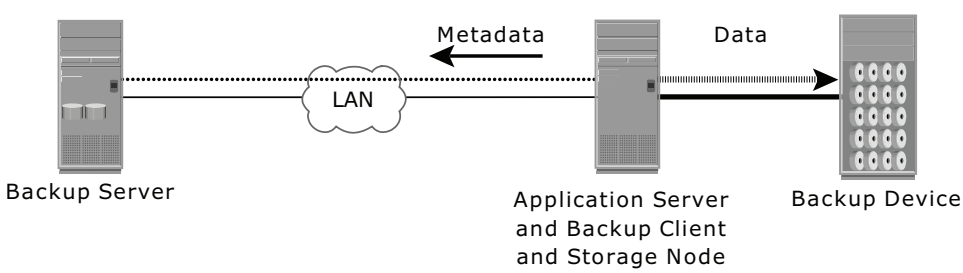

Figure 12-7: Direct-attached backup topology

In *LAN-based backup*, all servers are connected to the LAN and all storage devices are directly attached to the storage node (see Figure 12-8). The data to be backed up is transferred from the backup client (source), to the backup device (destination) over the LAN, which may affect network performance. Streaming across the LAN also affects network performance of all systems connected to the same segment as the backup server. Network resources are severely constrained when multiple clients access and share the same tape library unit (TLU).

This impact can be minimized by adopting a number of measures, such as configuring separate networks for backup and installing dedicated storage nodes for some application servers.

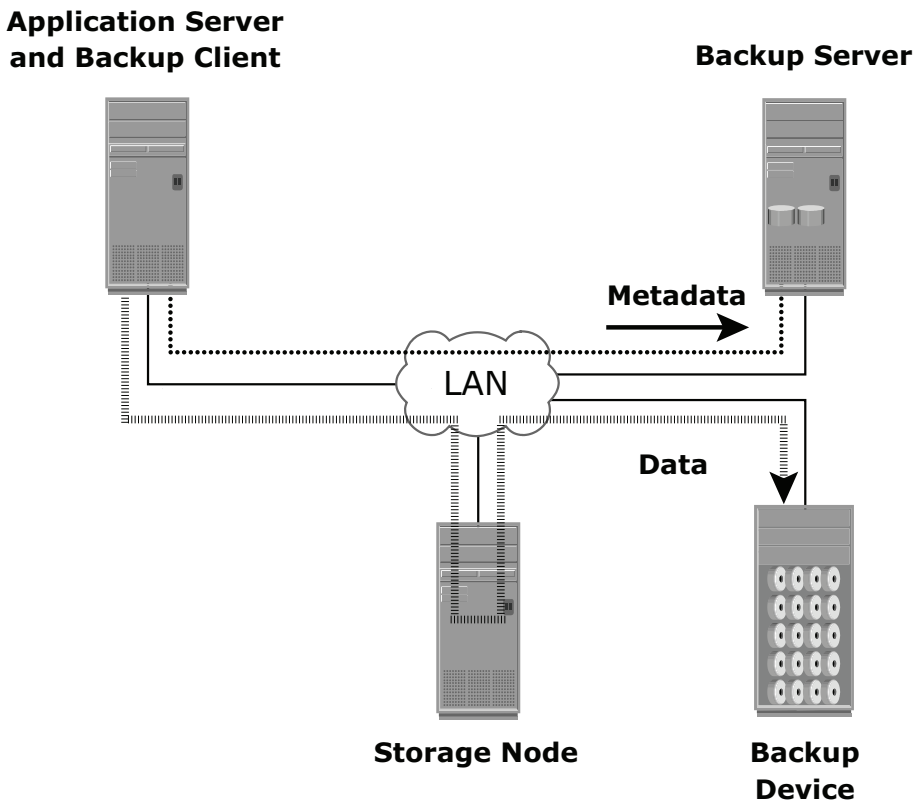

Figure 12-8: LAN-based backup topology

The *SAN-based backup* is also known as the *LAN-free backup*. Figure 12-9 illustrates a SAN-based backup. The SAN-based backup topology is the most appropriate solution when a backup device needs to be shared among the clients. In this case the backup device and clients are attached to the SAN.

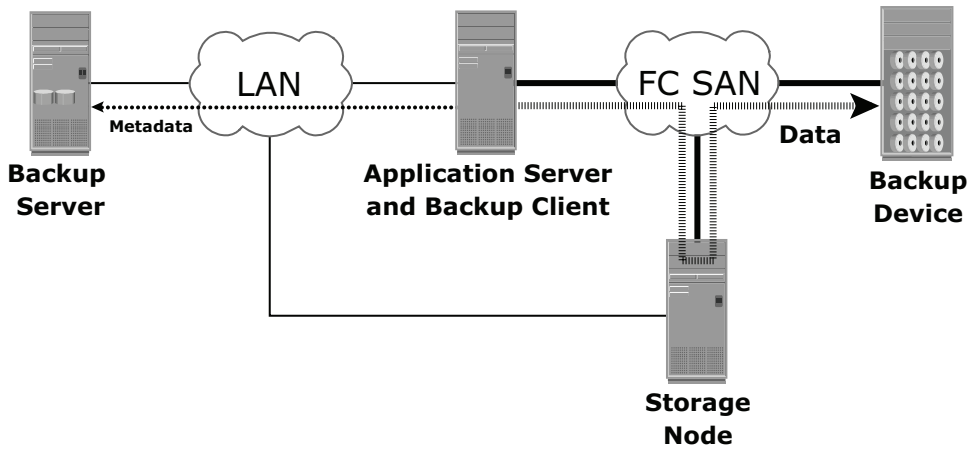

Figure 12-9: SAN-based backup topology

In this example, clients read the data from the mail servers in the SAN and write to the SAN attached backup device. The backup data traffic is restricted to the SAN, and backup metadata is transported over the LAN. However, the volume of metadata is insignificant when compared to production data. LAN performance is not degraded in this configuration.

By removing the network bottleneck, the SAN improves backup to tape performance because it frees the LAN from backup traffic. At the same time, LAN-free backups may affect the host and the application, as they consume host I/O bandwidth, memory, and CPU resources.

The emergence of low-cost disks as a backup medium has enabled disk arrays to be attached to the SAN and used as backup devices. A tape backup of these data backups on the disks can be created and shipped offsite for disaster recovery and long-term retention.

The *mixed topology* uses both the LAN-based and SAN-based topologies, as shown in Figure 12-10. This topology might be implemented for several reasons, including cost, server location, reduction in administrative overhead, and performance considerations.

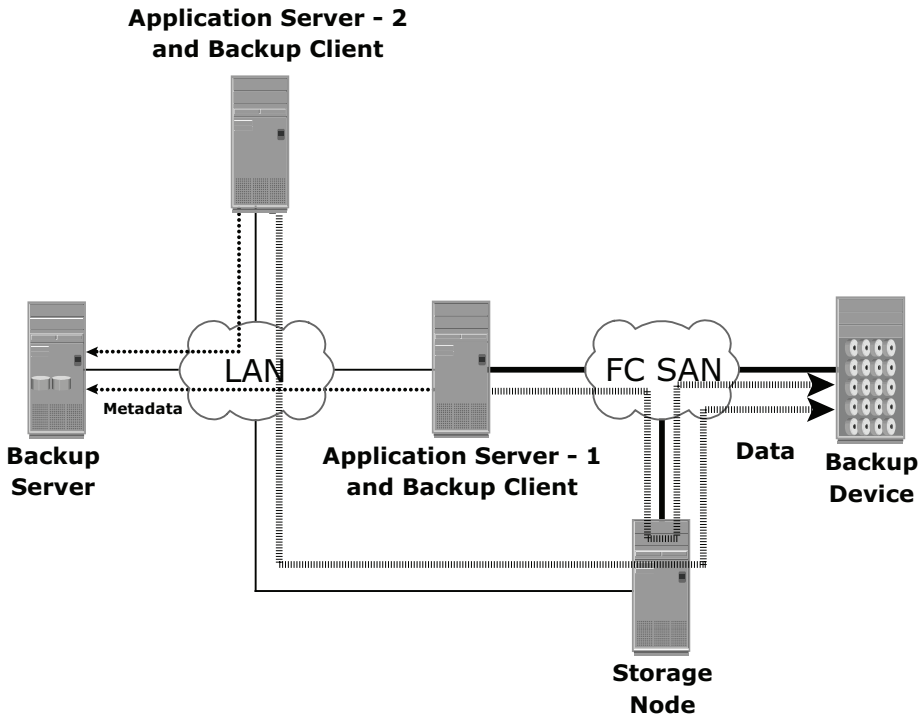

Figure 12-10: Mixed backup topology

12.8.1 Serverless Backup

Serverless backup is a LAN-free backup methodology that does not involve a backup server to copy data. The copy may be created by a network-attached controller, utilizing a SCSI extended copy or an appliance within the SAN. These backups are called serverless because they use SAN resources instead of host resources to transport backup data from its source to the backup device, reducing the impact on the application server.

Another widely used method for performing serverless backup is to leverage local and remote replication technologies. In this case, a consistent copy of the production data is replicated within the same array or the remote array, which can be moved to the backup device through the use of a storage node. Replication technologies are covered in detail in Chapter 13 and Chapter 14.

12.9 Backup in NAS Environments

The use of NAS heads imposes a new set of considerations on the backup and recovery strategy in NAS environments. NAS heads use a proprietary operating system and file system structure supporting multiple file-sharing protocols.

In the NAS environment, backups can be implemented in four different ways: server based, serverless, or using Network Data Management Protocol (NDMP) in either NDMP 2-way or NDMP 3-way.

In *application server-based backup*, the NAS head retrieves data from storage over the network and transfers it to the backup client running on the application server. The backup client sends this data to a storage node, which in turn writes the data to the backup device. This results in overloading the network with the backup data and the use of production (application) server resources to move backup data. Figure 12-11 illustrates server-based backup in the NAS environment.

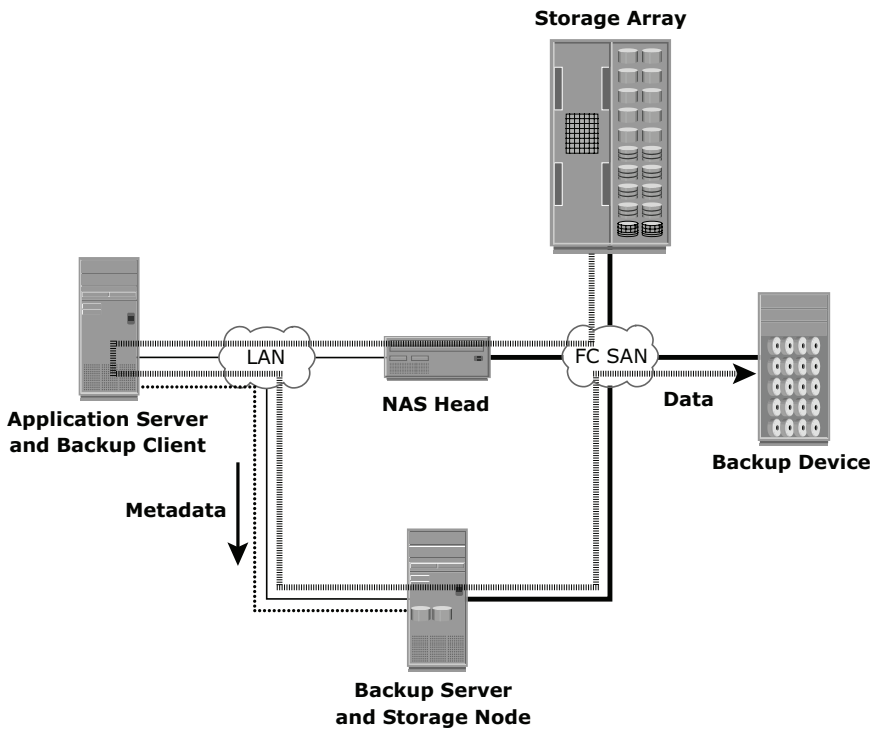

Figure 12-11: Server-based backup in NAS environment

In *serverless backup*, the network share is mounted directly on the storage node. This avoids overloading the network during the backup process and eliminates the need to use resources on the production server. Figure 12-12 illustrates serverless backup in the NAS environment. In this scenario, the storage node, which is also a backup client, reads the data from the NAS head and writes it to the backup device without involving the application server. Compared to the previous solution, this eliminates one network hop.

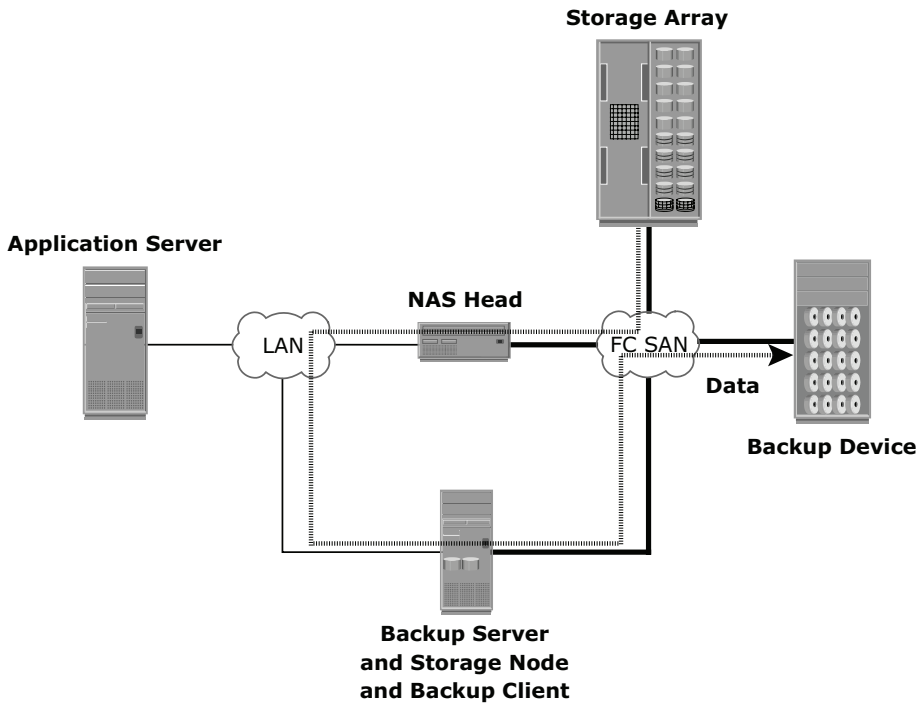

Figure 12-12: Serverless backup in NAS environment

NETWORK DATA MANAGEMENT PROTOCOL (NDMP)

NDMP is an industry-standard TCP/IP-based protocol for moving data across the network for backup and recovery. It can communicate with several interfaces for data transfer and enables vendors to use a common protocol for the backup architecture. Data can be backed up using NDMP regardless of the operating system or platform. Due to its flexibility, it is no longer necessary to transport data through the backup server, which reduces the load on the backup server and improves backup speed.

NDMP manages and transports data for robotics control of the tape library on the network. It optimizes backup and restore speeds due to the high-speed connection between the tape library and the NAS head.

In NDMP, backup data is sent directly from the NAS head to the backup device, while metadata is sent to the backup server. Figure 12-13 illustrates

backup in the NAS environment using NDMP 2-way. In this model, network traffic is minimized by isolating data movement from the NAS head to the locally attached tape library. Only metadata is transported on the network. This backup solution meets the strategic need to centrally manage and control distributed data while minimizing network traffic.

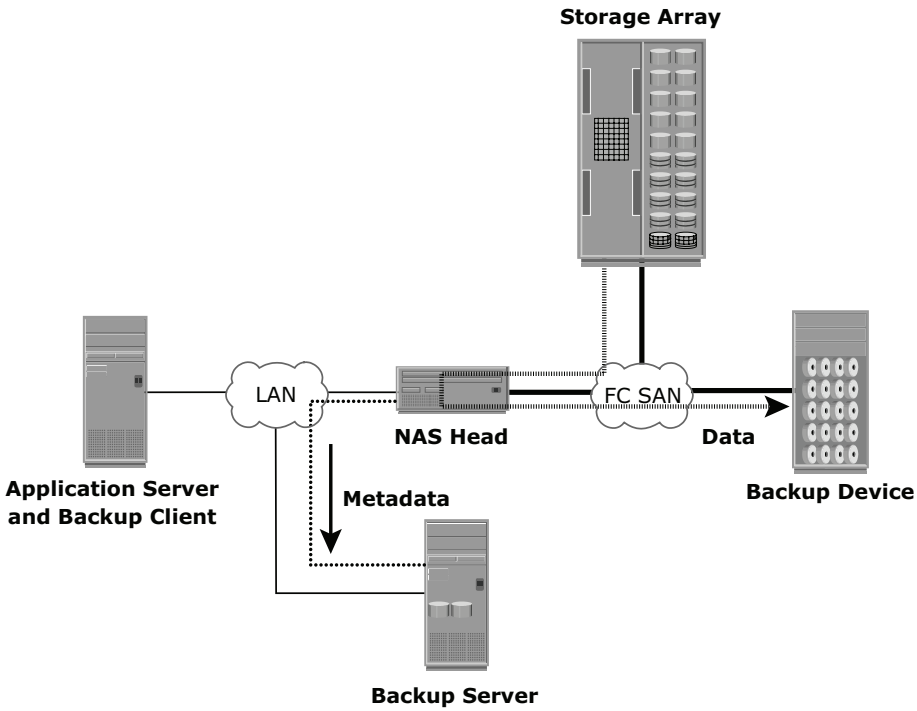

Figure 12-13: NDMP 2-way in NAS environment

In an *NDMP 3-way* file system, data is not transferred over the public network. A separate private backup network must be established between all NAS heads and the "backup" NAS head to prevent any data transfer on the public network in order to avoid any congestion or affect production operations. Metadata and NDMP control data is still transferred across the public network. Figure 12-14 depicts NDMP 3-way backup.

NDMP 3-way is useful when you have limited backup devices in the environment. It enables the NAS head to control the backup device and share it with other NAS heads by receiving backup data through NDMP.

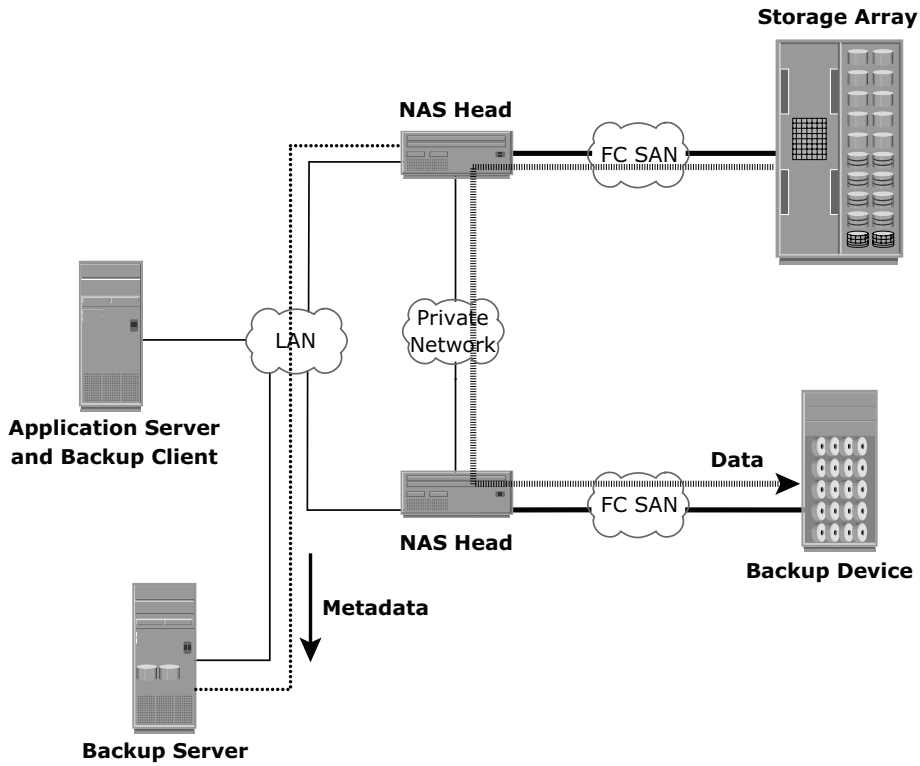

Figure 12-14: NDMP 3-way in NAS environment

12.10 Backup Technologies

A wide range of technology solutions are currently available for backup. Tapes and disks are the two most commonly used backup media. The tape technology has matured to scale to enterprise demands, whereas backup to disk is emerging as a viable option with the availability of low-cost disks. Virtual tape libraries use disks as backup medium emulating tapes, providing enhanced backup and recovery capabilities.

12.10.1 Backup to Tape

Tapes, a low-cost technology, are used extensively for backup. Tape drives are used to read/write data from/to a tape cartridge. Tape drives are referred to as sequential, or linear, access devices because the data is written or read sequentially.

Tape Mounting is the process of inserting a tape cartridge into a tape drive. The tape drive has motorized controls to move the magnetic tape around, enabling the head to read or write data.

Several types of tape cartridges are available. They vary in size, capacity, shape, number of reels, density, tape length, tape thickness, tape tracks, and supported speed. Today, a tape cartridge is composed of a magnetic tape with single or dual reels in a plastic enclosure.

A linear recording method was used in older tape drive technologies. This recording method consisted of data being written by multiple heads in parallel tracks, spanning the whole tape. Some tape drives used a helical scan method, which wrote the data diagonally. Modern tape drives use a linear serpentine method, which uses more tracks and fewer tape drive heads. Data is written in the same way as the linear method except that once the tape ends, the heads are moved and data continues to be written backward.

12.10.2 Physical Tape Library

The physical tape library provides housing and power for a number of tape drives and tape cartridges, along with a robotic arm or picker mechanism. The backup software has intelligence to manage the robotic arm and entire backup process. Figure 12-15 shows a physical tape library.

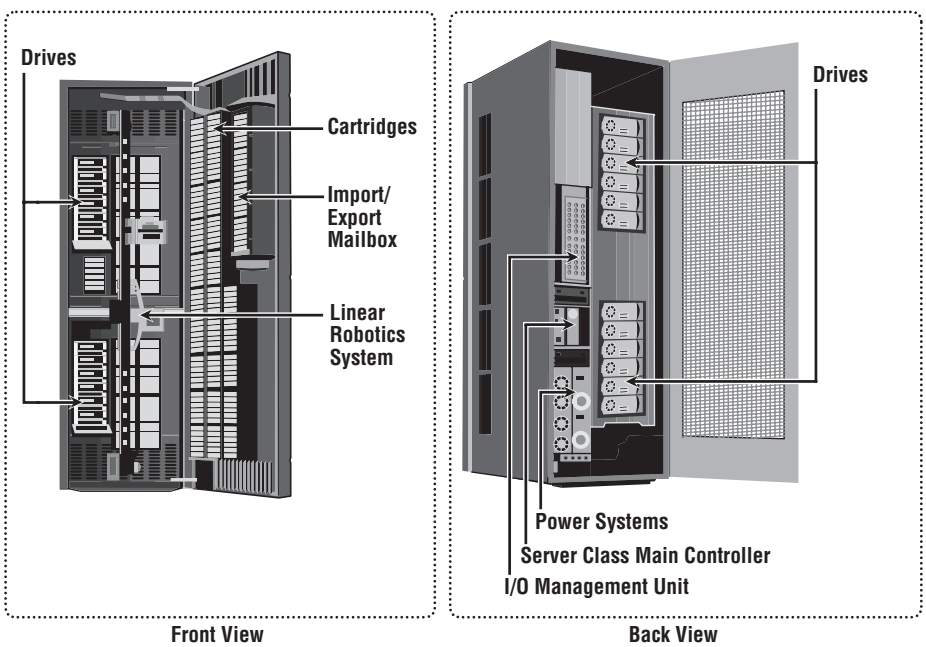

Figure 12-15: Physical tape library

Tape drives read and write data from and to a tape. Tape *cartridges* are placed in the *slots* when not in use by a tape drive. *Robotic arms* are used to move tapes around the library, such as moving a tape drive into a slot. Another type of slot called a *mail* or *import/export slot* is used to add or remove tapes from the library without opening the access doors (refer to Figure 12-15 Front View) because opening the access doors causes a library to go offline. In addition, each physical component in a tape library has an individual *element address* that is used as an addressing mechanism for moving tapes around the library.

When a backup process starts, the robotic arm is instructed to load a tape to a tape drive. This process adds to the delay to a degree depending on the type of hardware used, but it generally takes 5 to 10 seconds to mount a tape. After the tape is mounted, additional time is spent to position the heads and validate header information. This total time is called *load to ready time,* and it can vary from several seconds to minutes. The tape drive receives backup data and stores the data in its internal buffer. This backup data is then written to the tape in blocks. During this process, it is best to ensure that the tape drive is kept busy continuously to prevent gaps between the blocks. This is accomplished by buffering the data on tape drives. The speed of the tape drives can also be adjusted to match data transfer rates.

Tape drive *streaming* or *multiple streaming* writes data from multiple streams on a single tape to keep the drive busy. Shown in Figure 12-16, multiple streaming improves media performance, but it has an associated disadvantage. The backup data is interleaved because data from multiple streams is written on it. Consequently, the data recovery time is increased.

Figure 12-16: Multiple streams on tape media

Many times, even the buffering and speed adjustment features of a tape drive fail to prevent the gaps, causing the "shoe shining effect." This results in the tape drive stopping and rewinding to the appropriate point. The tape drive resumes writing only when its buffer is full, adversely affecting backup performance. When the tape operation is complete, the tape rewinds to the starting position and it is unmounted. The robotic arm is then instructed to move the unmounted tape back to the slot. *Rewind time* can range from several seconds to minutes.

When a *restore* is initiated, the backup software identifies which tapes are required. The robotic arm is instructed to move the tape from its slot to a tape drive. If the required tape is not found in the tape library, the backup software displays a message, instructing the operator to manually insert the required tape in the tape library. When a file or a group of files require restores, the tape must move sequentially to the beginning of the data before it can start reading. This process can take a significant amount of time, especially if the required files are recorded at the end of the tape.

Modern tape devices have an indexing mechanism that enables a tape to be fast forwarded to a location near the required data. The tape drive then fine-tunes the tape position to get to the data. However, before adopting a solution that uses tape drives supporting this mechanism, one must consider the benefits of data streaming performance against the cost of writing an index.

Limitations of Tape

Tapes are primarily used for long-term offsite storage because of their low cost. Tapes must be stored in locations with a controlled environment to ensure preservation of the media and prevent data corruption. Data access in a tape is sequential, which can slow backup and recovery operations. Physical transportation of the tapes to offsite locations also adds management overhead.

12.10.3 Backup to Disk

Disks have now replaced tapes as the primary device for storing backup data because of their performance advantages. Backup-to-disk systems offer ease of implementation, reduced cost, and improved quality of service. Apart from performance benefits in terms of data transfer rates, disks also offer faster recovery when compared to tapes.

Backing up to disk storage systems offers clear advantages due to their inherent random access and RAID-protection capabilities. In most backup environments, backup to disk is used as a staging area where the data is copied temporarily before transferring or staging it to tapes later. This enhances backup performance. Some backup products allow for backup images to remain on the disk for a period of time even after they have been staged. This enables a much faster restore. Figure 12-17 illustrates a recovery scenario comparing tape versus disk in a Microsoft Exchange environment that supports 800 users with a 75 MB mailbox size and a 60 GB database. As shown in the figure, a restore from disk took 24 minutes compared to the restore from a tape, which took 108 minutes for the same environment.

Recovery with a local replica, a full backup copy stored on disk and kept onsite, provides the fastest recovery solution. Using a disk enables the creation of full backups more frequently, which in turn improves RPO.

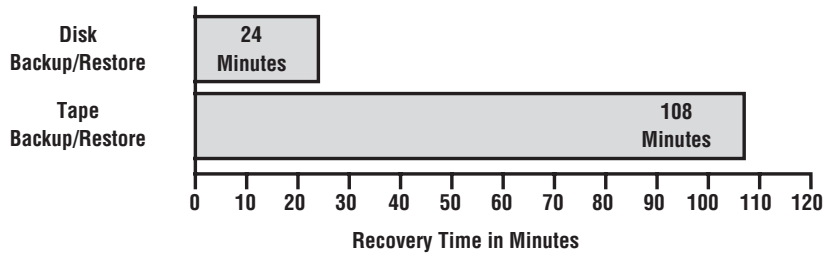

Figure 12-17: Tape versus disk restore

Backup to disk does not offer any inherent offsite capability, and is dependent on other technologies such as local and remote replication. In addition, some backup products require additional modules and licenses to support backup to disk, which may also require additional configuration steps, including creation of RAID groups and file system tuning. These activities are not usually performed by a backup administrator. Using backup to disk requires administrators to be aware of the file system's structure, fragmentation, file sizes, file system types, block size, and caching. The way the backup application interacts with the file system affects the way backup and restore occur.

For example, some backup products use very deep directory structures and group backup data in pre-allocated large files, reducing fragmentation. Other backup products use a "flat" structure, creating the backup files while the backup is running. This increases fragmentation, as more and more backup files are written to the same directory level. Therefore, it is important to understand how specific backup software operates and use the best practices provided by the storage manufacturer and the backup software vendor.

12.10.4 Virtual Tape Library

A *virtual tape library (VTL)* has the same components as that of a physical tape library except that the majority of the components are presented as virtual resources. For backup software, there is no difference between a physical tape library and a virtual tape library. Figure 12-18 shows a virtual tape library.

Virtual tape libraries use disks as backup media. Emulation software has a database with a list of virtual tapes, and each virtual tape is assigned a portion of a LUN on the disk. A virtual tape can span multiple LUNs if required. File

system awareness is not required while using backup to disk because virtual tape solutions use raw devices.

Similar to a physical tape library, a robot mount is performed when a backup process starts in a virtual tape library. However, unlike a physical tape library, where this process involves some mechanical delays, in a virtual tape library it is almost instantaneous. Even the *load to ready* time is much less than in a physical tape library.

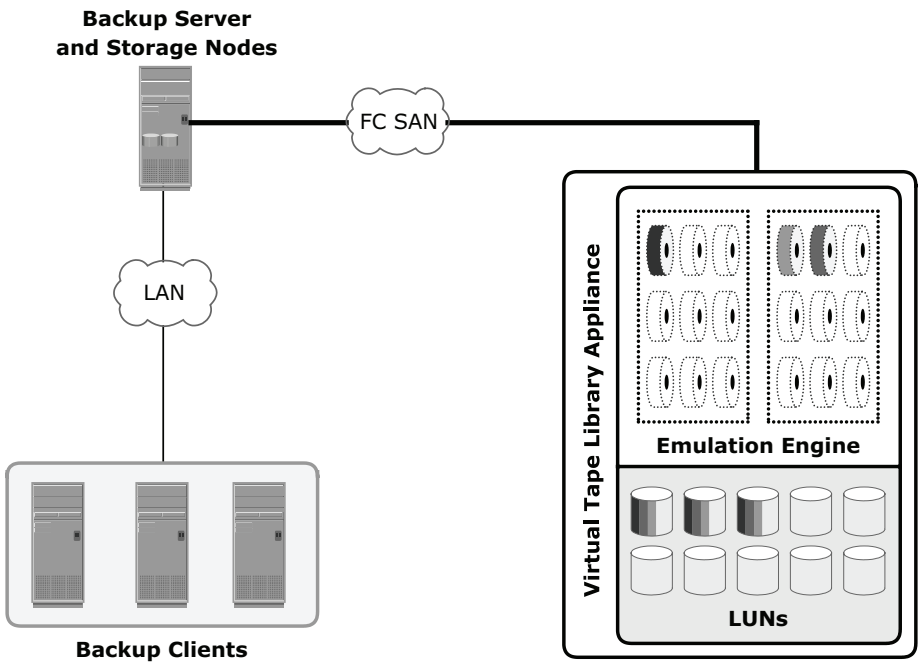

Figure 12-18: Virtual tape library

After the virtual tape is mounted and the tape drive is positioned, the virtual tape is ready to be used, and backup data can be written to it. Unlike a physical tape library, the virtual tape library is not constrained by the shoe shining effect. In most cases, data is written to the virtual tape immediately. When the operation is complete, the backup software issues a rewind command and then the tape can be unmounted. This rewind is also instantaneous. The virtual tape is then unmounted, and the virtual robotic arm is instructed to move it back to a virtual slot.

The steps to restore are similar to those in a physical tape library, but the restore operation is instantaneous. Even though virtual tapes are based on disks, which provide random access, they still emulate the tape behavior.

Virtual tape library appliances offer a number of features that are not available with physical tape libraries. Some virtual tape libraries offer *multiple*

emulation engines configured in an active cluster configuration. An engine is a dedicated server with a customized operating system that makes physical disks in the VTL appear as tapes to the backup application. With this feature, one engine can pick up the virtual resources from another engine in the event of any failure and enable the clients to continue using their assigned virtual resources transparently.

Replication over IP is available with most of the virtual tape library appliances. This feature enables virtual tapes to be replicated over an inexpensive IP network to a remote site. As a result, organizations can comply with offsite requirements for backup data. It's also possible to connect the engines of a virtual tape library appliance to a physical tape library, enabling the virtual tapes to be copied onto the physical tapes, which can then be sent to a vault or shipped to an offsite location.

Using virtual tape offers several advantages over both physical tapes and disks. Compared to physical tape, virtual tape offers better single stream performance, better reliability, and random disk access characteristics. Backup and restore operations are sequential by nature, but they benefit from the disk's random access characteristics because they are always online and ready to be used, improving backup and recovery times. Virtual tape does not require the usual maintenance tasks associated with a physical tape drive, such as periodic cleaning and drive calibration. Compared to backup-to-disk devices, virtual tapes offer easy installation and administration and inherent offsite capabilities. In addition, virtual tapes do not require any additional modules or changes on the backup software.

However, virtual tapes can be used only for backup purposes because they are usually offered as an appliance. In a backup-to-disk environment, the disk systems can be used for both production and backup data. Virtual tape appliances are preconfigured from the manufacturer, facilitating easy installation and administration.

Table 12-1 shows a comparison between various backup technology options.

Table 12-1: Backup Technology Comparison

FEATURES	TAPE	DISK-AWARE BACKUP-TO-DISK	VIRTUAL TAPE
Offsite Capabilities	Yes	No	Yes
Reliability	No inherent protection methods	Yes	Yes
Performance	Subject to mechanical operations, load times	Faster single stream	Faster single stream
Use	Backup only	Multiple (backup/production)	Backup only

12.11 Concepts in Practice: EMC NetWorker

EMC backup products provide a powerful and effective way to back up and recover data. This ensures higher information protection and enables compliance with regulations and corporate policies. The EMC backup recovery portfolio consists of a broad range of products for an ever-increasing amount of backup data that presents a challenge to organizations such as demands of shorter backup windows, quicker restore responses, and de-duplication of backup data. EMC's backup products help organizations to meet these challenges through software and disk-based technologies. This section provides details of EMC NetWorker and a brief introduction about various other backup and recovery products, their major features, functionality, terminology, and processes.

Traditionally, the industry used tape backups that follow a one-size-fits-all strategy. However, tapes are challenged to meet service-level requirements. EMC NetWorker provides the capability to use disks for backup instead of tapes. The advanced backup capabilities enable the use of array-based snapshot and replication technology. These features of EMC NetWorker ensure high performance for backup and recovery. Visit `http://education.EMC.com/ismbook` for the latest information.

NetWorker enables simultaneous-access operations to a volume, for both reads and writes, as opposed to a single operation with tapes. NetWorker works within the existing framework of the hardware, operating system, software, and network communication protocols to provide protection for critical business data by centralizing, automating, and accelerating backup and recovery operations across an enterprise.

A single NetWorker server can be used to protect all clients and servers in the backup environment. NetWorker provides support for heterogeneous storage devices and platforms, and it integrates with popular databases and applications. NetWorker supports clustering technologies and open-file backup, and is compatible with tapes, disks, and virtual tape libraries. It uses the client/server model, which distributes the workload, improves backup performance, and provides network-based backup protection.

NetWorker provides advanced backup capability that leverages disk-based technologies, such as instant restore, off-host backups, and integration of backup with snapshots and full-volume mirrors. It enables meeting stringent RTO and RPO requirements.

NetWorker provides cold and hot backups, and supports a wide range of applications for hot backups with granular-level recovery. In a hot backup, NetWorker extracts data for backup using an API, and the application remains open even during the backup.

It uses 256-bit AES (advanced encryption standard) encryption authentication to provide increased security for communication between the hosts. Host machines are authenticated using the Secure Sockets Layer (SSL) protocol and self-signed certificates.

NetWorker also provides centralized management of the backup environment through a GUI, customizable reporting, and wizard-driven configuration. With the NetWorker Management Console (NMC), it can be easily administered from any host with a supported Web browser. NetWorker also provides many command-line utilities. With its various configuration points, NetWorker can be customized to meet the backup requirements of specific organizational needs. To facilitate NetWorker administration, several reports are available through the NMC reporting feature. Data maintained in the NMC server database, gathered from any or all of the NetWorker servers, is used to prepare reports on backup statistics and status, events, hosts, users, and devices.

NetWorker clients generate backups called *save sets*. NetWorker can back up multiple save sets from clients running different operating systems to any NetWorker-configured device. NetWorker can write more than one save set to a storage volume, and it supports backup to multiple devices that may be located at remote sites.

NetWorker also supports Open Tape Format (OTF), a data format that enables multiplexed, heterogeneous data to reside on the same tape. Using OTF, a NetWorker storage node can be migrated to a host running a different operating system.

12.11.1 NetWorker Backup Operation

In a NetWorker backup operation (refer to Figure 12-19), the NetWorker client pushes the backup data to the destination storage node. The client generates tracking information, including the file and directory names in the backup and the time of the backup, and sends it to the server to facilitate point-in-time recoveries. The storage node organizes the client's data and writes it to backup devices. Storage nodes also send tracking information about the save sets written during the backup to the NetWorker server. NetWorker also enables automating and scheduling the backup process.

NetWorker can initiate backup in two ways: client-initiated and server-initiated. A *client-initiated backup* is a manual/automated process that is initiated by a NetWorker client, whereas a *server-initiated backup* is initiated from the NetWorker server. The NetWorker server sends a backup request to one or more NetWorker clients. A server-initiated backup is usually configured to start automatically, but it can also be performed manually. This can be done from the NetWorker administration window or the CLI.

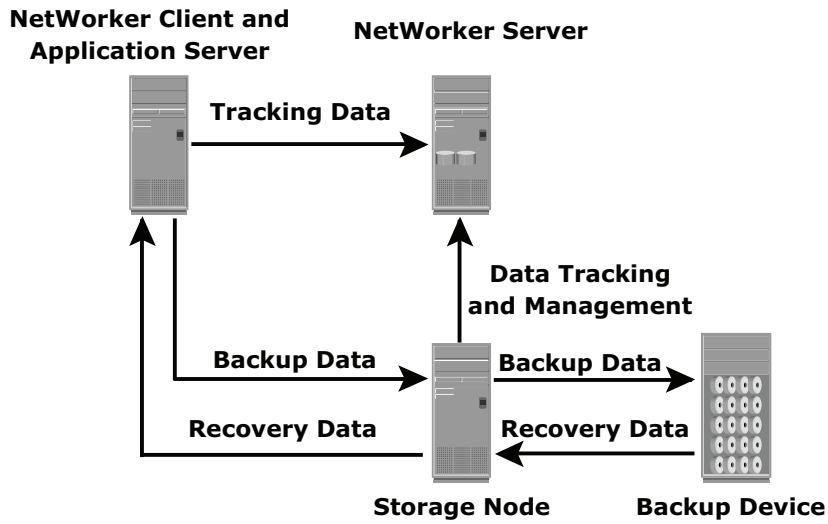

Figure 12-19: NetWorker operation

12.11.2 NetWorker Recovery

NetWorker is flexible in the way recovery operations are performed, and it maintains security to avoid recovery of data by unauthorized users. Recoverable data can include files, directories, file systems, or application data. Files can be recovered to a directory other than the directory from which they were backed up. NetWorker detects, and can be configured to automatically resolve, naming conflicts. The three types of manual recoveries — browsable, save set, and directed — are all processes initiated from a NetWorker client.

12.11.3 EmailXtender

EmailXtender is a comprehensive archive application that automatically collects, organizes, retains, and retrieves e-mail messages and attachments. It works with all major messaging environments, including Microsoft Exchange/Outlook and Lotus Domino/Notes. EmailXtender copies messages from mail servers into a dedicated archive by performing journaling on the e-mail server, preventing them from growing large enough to impact e-mail system performance. EmailXtender provides plug-in and web-based search facilities that improve storage management and operational efficiency of the messaging environment. With EmailXtender, companies can automatically enforce retention policies by periodically deleting messages from mail servers and archiving them to EmailXtender. The EmailXtract feature of EmailXtender enables removing messages from the mail server and replacing them with pointers or shortcuts to copies of the

messages archived in EmailXtender. Users can seamlessly access these messages as if they were still stored on the mail server. This improves e-mail server performance, backup, and recovery performance, and reduces storage costs.

12.11.4 DiskXtender

DiskXtender is a robust storage management solution available only for the Microsoft Windows platform. DiskXtender extends the amount of space on the local NTFS (NT file system) volume. It does this by migrating files from the local drive to external media and purging files from primary storage. To a client retrieving files from the drive extended by DiskXtender, all files, whether on the extended drive or on storage media, appear to be present locally. Transfer of files to secondary storage can be controlled by a set of rules that detail migration criteria, such as a file's age, size, type, or other attributes. To avoid media-swapping delays during data request, DiskXtender intelligently queues requests for files and accesses secondary media only when necessary. To enhance availability, it can migrate a single file to as many as four different targets. Clients writing to the extended drive are unaware of the volume being extended by DiskXtender. Writing a file to a secondary media adds extended attribute information to the file on the extended drive. Purging a file removes the file data from the extended drive, leaving behind a file tag on the drive. When a file on the extended drive is requested, it is provided directly by the operating system; but if the file isn't found on the extended drive, the request is forwarded to the media storage software, which locates the appropriate media, and locates the file.

12.11.5 Avamar

Avamar is a comprehensive, client-server network backup and restore solution. Avamar differs from traditional backup and restore solutions by identifying and storing only unique sub-file data objects. Redundant data is identified at the source, drastically reducing the amount of data that travels across the network to be stored and managed by the backup host. Avamar also creates and stores "trees" that link all data objects from a single backup. These trees are used to recreate files for restore. Avamar uses standard IP network technology, so dedicated backup networks are not required. During backup, the Avamar client traverses each directory and examines the local cache to determine which files have not been previously backed up. Once an object is backed up on the server, it is never sent for backup again. This drastically reduces network traffic and enhances backup storage efficiency, guaranteeing the most effective de-duplication of the data. After an object has been stored, it cannot be deleted until the specified retention period has expired.

12.11.6 EMC Disk Library (EDL)

EDL is a dedicated virtual tape library appliance. It consists of a storage system and a Linux-based server. The server runs software that emulates tape drives and tape libraries. To the backup software, the virtual tape drives and cartridges will look and behave like physical libraries, drives, and cartridges.

Summary

Data availability is a critical requirement for information-centric businesses. Backups protect businesses from data loss and also helps to meet regulatory and compliance requirements.

This chapter detailed backup considerations, methods, technologies, and implementations in a storage networking environment. It also elaborated various backup topologies and architectures.

Although the selection of a particular backup media is driven by the defined RTO and RPO, disk-based backup has a clear advantage over tape-based backup in terms of performance, availability, faster recovery, and ease of management. These advantages are further supplemented with the use of replication technologies to achieve the highest level of service and availability requirements. Replication technologies are covered in detail in the next two chapters.

EXERCISES

1. A manufacturing corporation uses tape as its primary backup storage media throughout the organization. Full backups are performed every Sunday. Incremental backups are performed Monday through Saturday. The environment contains many backup servers, backing up different groups of servers.
   - The e-mail and database applications have to be shut down during the backup process. Due to the decentralized backup environment, recoverability is often compromised. There are too many tapes that need to be mounted to perform a full recovery in case of a complete failure. The time needed to recover is too lengthy.
   - The company would like to deploy an easy-to-manage backup environment. They want to reduce the amount of time the e-mail and database applications are unavailable, and reduce the number of tapes required to fully recover a server in case of a failure.
   - Propose a backup and recovery solution to address the company's needs. Justify how your solution ensures that their requirements will be met.
2. There are limited backup devices in a file sharing NAS environment. Suggest a suitable backup implementation that will minimize the network traffic, avoid any congestion, and at the same time not impact the production operations. Justify your answer.
3. Discuss the security concerns in backup environment.
4. What are the various business/technical considerations for implementing a backup solution, and how do these considerations impact the backup solution/implementation?
5. What is the purpose of performing operation backup, disaster recovery, and archiving?
6. List and explain the considerations in using tape as the backup technology. What are the challenges in this environment?
7. Describe the benefits of using "virtual tape library" over "physical tapes."